Technische Universität München
TUM School of Computation, Information and Technology

Computational Modeling of Protein-RNA Interactions with Deep Neural Networks

Marc Joel Horlacher

Vollständiger Abdruck der von der *TUM School of Computation, Information and Technology* der Technischen Universität München zur Erlangung eines

Doktors der Naturwissenschaften (Dr. rer. nat.)

genehmigten Dissertation.

Vorsitz: Prof. Dr. Nassir Navab

Prüfende der Dissertation:

1. Prof. Dr. Julien Gagneur
2. Prof. Dr. Dr. Fabian J. Theis
3. Prof. Dr. Eduardo Eyras

Die Dissertation wurde am 12.12.2023 bei der Technischen Universität München eingereicht und durch die TUM School of Computation, Information and Technology am 30.07.2024 angenommen.

COMPUTATIONAL MODELING OF PROTEIN-RNA INTERACTIONS WITH DEEP NEURAL NETWORKS

MARC HORLACHER

ABSTRACT

Post-transcriptional processes are governed by the intricate interplay between RNA molecules and RNA-binding proteins, dictating the journey of RNA from transcription to translation and eventual decay. Unraveling the mechanisms underlying these interactions is crucial for understanding their role in gene regulatory processes, elucidating their relevance in disease contexts, and identifying novel therapeutic targets. Experimental techniques, such as in *in vivo* CLIP, provide the means to profile protein-RNA interactions transcriptome-wide. The analysis of CLIP data often involves the training of machine learning models. These models not only facilitate the prediction of protein-RNA interactions for uncharacterized RNA sequences but also enable the discovery of RNA binding motifs through model explainability techniques and variant impact scoring via *in silico* perturbation analyses. This doctoral thesis is concerned with the computational modeling of protein-RNA interactions using deep learning methods. Following a comprehensive review of both biological and computational fundamentals, it contributes three major scientific advancements. First, it utilizes deep learning models trained on a large CLIP dataset from human cell lines to create a computational binding map of human proteins with the SARS-CoV-2 RNA, providing insights into critical protein-RNA interactions in the context of viral infection. Next, it presents a benchmark study that evaluates various deep learning methods for the prediction of protein-RNA interactions, enhancing our understanding of the most effective approaches for this crucial task. Finally, it introduces a novel computational model capable of predicting protein-RNA interactions at nucleotide resolution, handling sequences of arbitrary lengths, scoring the impact of sequence variants, and identifying essential binding motifs. Collectively, these contributions advance our knowledge of protein-RNA interactions, their role in the regulation of post-transcriptional processes and human disease.

ZUSAMMENFASSUNG

Post-transkriptionelle Prozesse werden durch das komplexe Zusammenspiel zwischen RNA-Molekülen und RNA-bindenden Proteinen bestimmt, die den Weg der RNA von der Transkription über die Translation bis zu ihrem Zerfall kontrollieren. Das Entschlüsseln der Mechanismen, die diesen Interaktionen zugrunde liegen, ist von entscheidender Bedeutung für das Verständnis ihrer Rolle bei der Genregulation, ihrer Bedeutung im Zusammenhang mit Krankheiten, sowie für die Identifizierung potenzieller neuer therapeutischer Ziele. Experimentelle Methoden, wie *in vivo* CLIP, bieten die Möglichkeit, Protein-RNA-Interaktionen transkriptomweit nachzuweisen. Die Analyse von CLIP-Daten beinhaltet häufig das trainieren von Modellen mittels maschinellem Lernens. Diese Modelle ermöglichen nicht nur die Vorhersage von Protein-RNA-Interaktionen für uncharakterisierte RNA-Sequenzen, sondern auch das Bestimmen von RNA-Bindemotiven und das Quantifizieren der Auswirkungen von Mutationen. Diese Doktorarbeit befasst sich mit der computergestützten Modellierung von Protein-RNA-Interaktionen mittels Deep-Learning-Methoden. Neben einem umfassenden Überblick über biologische und informatische Grundlagen besteht sie aus drei wichtige wissenschaftliche Beiträge. Erstens werden Deep-Learning-Modelle verwendet, welche auf einem großen CLIP-Datensatz von menschlichen Zelllinien trainiert wurden, um Interaktionen menschlicher Proteine mit der SARS-CoV-2 RNA vorherzusagen. Dies ermöglicht Einblicke in kritische Protein-RNA-Interaktionen im Kontext einer SARS-CoV-2-Infektion. Als Nächstes wird eine Benchmark-Studie vorgestellt, in der verschiedene Deep-Learning-Methoden für die Vorhersage von Protein-RNA-Interaktionen evaluiert und die performantesten Ansätze für diese Aufgabe identifiziert werden. Schließlich wird ein neuartiges Deep-Learning Modell präsentiert, welches es erlaubt Protein-RNA-Interaktionen mit Nukleotidauflösung zu modellieren, Sequenzen beliebiger Länge zu verarbeiten, die Auswirkungen von Mutationen vorherzusagen und wesentliche Bindungsmotive zu identifizieren. Zusammengenommen erweitern diese Beiträge das Wissen über Protein-RNA-Interaktionen, sowie deren Rolle bei der Regulierung post-transkriptioneller Prozesse und im Kontext menschlicher Krankheiten.

ACKNOWLEDGMENTS

I am deeply appreciative of the individuals who have accompanied me on this transformative PhD journey - my colleagues, advisors, collaborators, and friends. In particular, I want to express my heartfelt gratitude to my advisor Dr. Annalisa Marsico, who guided me throughout my PhD. Her mentorship not only provided direction but also granted me the independence to explore and develop my own research projects and ideas. In addition, I'm expressing my gratitude to the members of the entire Computational RNA Biology group for their expertise and the warm atmosphere they fostered. In particular, I would like to thank my colleague Dr. Lambert Moyon, a true collaborator in every sense. Countless discussions with Lambert have enriched my academic experience, and his support and assistance across several research projects have been invaluable. Furthermore, I want to express my gratitude to Prof. Ole Winther for his generosity in hosting me during a significant six-month research stay at the University of Copenhagen. His expertise has been invaluable, and our regular meetings have provided valuable insights. None the least, I want to thank the brilliant students that I've had the pleasure to mentor during my PhD for their dedication and their valuable contributions to our research endeavors. Finally, I extend my sincere thanks to all those who have played a role, big or small, in the successful completion of my doctoral research. Your support has left a mark on both my academic journey and personal development.

PUBLICATIONS

This thesis is based on the following peer-reviewed core publication:

[1] Horlacher, Marc, Svitlana Oleshko, Yue Hu, Mahsa Ghanbari, Giulia Cantini, Patrick Schinke, Ernesto Elorduy Vergara et al. "A computational map of the human-SARS-CoV-2 protein–RNA interactome predicted at single-nucleotide resolution." NAR Genomics and Bioinformatics 5, no. 1 (2023): lqad010. [1]

[2] Horlacher, Marc, Giulia Cantini, Julian Hesse, Patrick Schinke, Nicolas Goedert, Shubhankar Londhe, Lambert Moyon, and Annalisa Marsico. "A systematic benchmark of machine learning methods for protein–RNA interaction prediction." Briefings in Bioinformatics 24, no. 5 (2023): bbad307. [2]

[3] Horlacher, Marc, Nils Wagner, Lambert Moyon, Klara Kuret, Nicolas Goedert, Marco Salvatore, Jernej Ule, Julien Gagneur, Ole Winther, and Annalisa Marsico. "Towards in silico CLIP-seq: predicting protein-RNA interaction via sequence-to-signal learning." Genome biology 24, no. 1 (2023): 1-37. [3]

The following publications were produced during the course of the doctoral studies, but are not subject to evaluation:

[1] Salvatore, Marco, Marc Horlacher, Annalisa Marsico, Ole Winther, and Robin Andersson. "Transfer learning identifies sequence determinants of cell-type specific regulatory element accessibility." NAR genomics and bioinformatics 5, no. 2 (2023): lqad026. [4]

[2] Wang, Jun, Marc Horlacher, Lixin Cheng, and Ole Winther. "RNA trafficking and subcellular localization—a review of mechanisms, experimental and predictive methodologies." Briefings in bioinformatics 24, no. 5 (2023): bbad249. [5]

[3] Schmidt, Nora, Sabina Ganskih, Yuanjie Wei, Alexander Gabel, Sebastian Zielinski, Hasmik Keshishian, Caleb A. Lareau et al. "SND1 binds SARS-CoV-2 negative-sense RNA and promotes viral RNA synthesis through NSP9." Cell (2023). [6]

CONTENTS

LIST OF FIGURES

ACRONYMS

APA Alternative Polyadenylation

ARE AU-rich Element

ASO Antisense Oligonucleotide

cDNA Complementary DNA

CDS Coding Sequence

circRNA circular RNA

CLIP Crosslinking-Immunoprecipitation

CNN Convolutional Neural Network

CPA Cleavage and Polyadenylation

DAG Directed Acyclic Graph

DNA Deoxyribonucleic acid

eCLIP Enhanced CLIP

eRIC enhanced RIC

eRIC Enhanced RIC

GELU Gaussian Error Linear Unit

iCLIP Individual-nucleotide CLIP

IDR Intrinsically Disordered Region

IG Integrated Gradients

IP Immunoprecipitation

KH K Homology

lncRNA long non-coding RNA

LSTM Long Short-Term Memory

m6A N6-methyladenosine

miCLIP m6A Individual-nucleotide CLIP

miRNA micro-RNA

MLP Multi-Layer Perceptron

mRNA messenger RNA

MS Mass Spectrometry

ncRNA non-coding RNA

ncRNA non-coding RNA

PAR-CLIP Photoactivatable Ribonucleoside-enhanced CLIP

PAS Poly-Adenylation Signal

PCR Polymerase Chain Reaction

PFM Position-frequency Matrix

PPI Protein-Protein-Interaction

PPM Position-probability Matrix

pre-mRNA precursor messenger RNA

PUM Pumilio Homology

PWM Position-weight Matrix

RAP RNA Antisense Purification

RBD RNA-binding Domain

RBP RNA-binding Protein

ReLU Rectified Linear Unit

RIC RNA Interactome capture

RIC RNA Interactome Capture

RNA Ribonucleic Acid

RNP Ribonucleoprotein Particle

RRM RNA Recognition Motif

rRNA ribosomal RNA

SGD Stochastic Gradient Descend

SGD Stochastic Gradient Descend

siRNA small interfering RNA

SMI Size-match Input

SNV Single-nucleotide Variant

ssRNA single-stranded RNA

SVM Support-Vector Machine

TF Transcription Factor

tRNA transfer RNA

UTR Untranslated Region

UV Ultraviolet

ZF Zinc Finger

Part I

INTRODUCTION

0.1 MOTIVATION

Computational modeling of protein-RNA interactions through machine learning is an active area of research. Knowledge of which RNA-binding Protein (RBP) binds to which RNA target can aid in characterizing the post-transcriptional trajectory of RNAs, quantifying the impact of their misregulation in human diseases, and identifying therapeutic targets. While substantial progress has been made, several open questions persist, particularly: How can these models be applied in practice to study interactions with exogenous RNA? What is the current landscape of state-of-the-art methods and which design choices are crucial for model performance? How can we enhance the performance and resolution of predictive methods? This thesis addresses these questions.

RBPs have been shown to play a significant role in viral infections, including SARS-CoV-2. However, while it has been demonstrated that the repression of host RBPs through pharmaceutical intervention can suppress viral replication [7], the specific functions of individual RBPs as either host or antiviral factors remain unclear [8]. Several studies identified a large number of human RBPs that interact with SARS-CoV-2 RNAs [7, 9]. However, these studies rely on RNA-centric methods that determine if — not where — an RNA is bound by a set of proteins. Interactions for only a fraction of RBPs have been profiled with high-resolution CLIP-based approaches, mainly due to cost [6, 7]. Yet, knowledge of precise RBP binding sites is essential to understand how SARS-CoV-2 hijacks human RBPs. To address this, this thesis investigates protein-RNA interaction sites of human RBPs with viral RNAs using high-resolution computational models [1].
Numerous machine learning methods for predicting protein-RNA interactions have been developed in recent years. While this has broadened the range of choices for researchers, evaluation of these methods across diverse datasets has posed challenges in discerning state-of-the-art approaches. It is evident that, to fully harness the recent surge in method development, a unified evaluation framework is imperative. Although such frameworks exist for other areas of genomics research, such as the ENCODE-DREAM challenge for transcription factor binding prediction [10, 11], a gap exists in the field of RBP-binding prediction research. To bridge this gap, this thesis introduces a comprehensive benchmark for protein-RNA interaction prediction methods [2].
Existing protein-RNA interaction prediction methods classify RNA sequence based on whether they are bound by a protein of interest. However, this binary approach presents several issues. First, it results in low-resolution predictions, as predicted labels are typically assigned to sequences that consist of hundreds of nucleotides, while the targets for most RNA-binding domains are significantly shorter. Second, the training of current models is highly sensitive to preprocessing steps that determine the set of bound sequences based on high-throughput data, such as CLIP [12]. Third, these methods may discard potentially informative nuances in the data generated by CLIP techniques and fail to harness the full potential of iCLIP or eCLIP experiments, which detect protein-RNA interactions at nucleotide resolution [13, 14]. Consequently, although classification performance of current methods is improving steadily, novel modeling approaches need to be devised to fully utilize the wealth of data produced by CLIP protocols. To this end, this thesis presents RBPNet, a method that predicts protein-RNA interactions at nucleotide resolution by directly modeling the raw CLIP data [3].

0.2 THESIS OUTLINE

This thesis is structured into three main parts. *Biological Background*, which establishes relevant preliminaries regarding RNA, RNA-binding proteins and experimental protocols. *Computational Background*, which introduces methodological concepts for the training and interpretation of (deep) neural networks and lastly *Academic Contributions*, which presents the three core publications. A detailed, chapter-wise outline of the thesis is given in the following.

Chapters 1 and 2 briefly review the role of RNA in cellular biology, with a focus on RNA processing and post-transcriptional gene regulation. It further offers an introduction to RNA-binding proteins and emphasizes their role in the context of RNA metabolism.

Chapter 3 presents an overview over experimental methods for measuring protein-RNA interaction, with a focus on protein-centric, CLIP-based approaches. It gives a detailed description of the eCLIP [14] protocol and outlines experimental biases commonly found in CLIP.

Chapter 4 covers essential preliminaries pertaining artificial neural networks. The chapter first introduces multi-layer perceptrons, followed by a brief description of common activation functions and an introduction to convolutional neural networks. The chapter concludes with an overview over model regularization techniques, including dropout and batch normalization, as well as residual connections.

Chapter 5 provides an introduction to model explainability for (deep) neural networks, with a focus on modeling tasks based on RNA and DNA sequences. The chapter introduces model-free approaches, such as feature perturbation or occlusion, as well as architecture-specific methods, including the direct interpretation of convolutional filters and attention maps. Lastly, the chapter gives a comprehensive description of gradient-based approaches, which have been used extensively in research related to this thesis, including [3] and [1].

Chapter 6 briefly reviews applications of deep learning to genomics and transcriptomics. It highlights recent trends towards genomic foundation models, such as large multi-task models and nucleotide language models.

Chapters 7, 8 and 9 present the three core publications of this thesis [1–3]. This includes a computational RBP-binding map of human RBPs to the SARS-CoV-2 genome, a systematic benchmark of classification-based deep learning methods for protein-RNA interaction prediction and a method for modeling protein-RNA interactions at single-nucleotide resolution, respectively.

Lastly, the academic contributions and future direction are discussed in the *Discussion and Outlook* part.

Part II

BIOLOGICAL BACKGROUND

1 THE RNA WORLD

Ribonucleic Acid (RNA) is a single-stranded molecule exhibiting remarkable complexity and versatility. It is composed of ribonucleotides Adenine (A), Guanine (G), Cytosine (C), and Uracil (U), as well as a number of non-canonical bases. Long considered a mere messenger between Deoxyribonucleic acid (DNA) and proteins, the understanding of RNA's diverse functions has expanded significantly in recent decades. Besides messenger RNA (mRNA), several forms of non-coding RNAs (ncRNAs) exist, including transfer RNAs (tRNAs), micro-RNAs (miRNAs), small interfering RNAs (siRNAs), ribosomal RNAs (rRNAs), long non-coding RNAs (lncRNAs) and circular RNAs (circRNAs), among others. ncRNAs have been shown to be involved in various gene regulatory processes, for instance, short miRNAs and siRNAs regulate gene expression by binding to target mRNAs and promote their degradation [15], while circRNAs may act as miRNA sponges [16]. Further, lncRNAs, which are usually over 200 nucleotides (nt) in length, play a role in chromatin remodeling [17] and the assembly of protein-RNA complexes [18] by acting as a scaffold. RNA can form sophisticated higher-order structures through Watson-Crick base-pairing, such as stem-loop structures or hairpins, enabling it to perform a variety of functions. For instance, cloverleaf-shaped tRNAs facilitate amino acids assembly by matching amino acids to their corresponding codon on the mRNA. In the case of *riboswitches*, which usually reside in Untranslated Regions (UTRs) of mRNAs, RNA structure can change dynamically upon encountering metabolites, thereby inducing pausing of transcription [19] and thus regulating gene expression. Owing to an RNAs function to both store genetic information as well as exhibit catalytic activity, RNA has been suggested to have been the primary molecule of an ancient *"RNA World"* [20]. Indeed, it has been shown that under certain conditions RNA may self-replicate, either via non-enzymatic template-directed polymerization or by a RNA polymerase ribozyme [21]. This highlights the central role of RNA in both the evolution of life on earth as well as the intricate molecular processes that govern gene regulation at the post-transcriptional level.

2 POST-TRANSCRIPTIONAL REGULATION

2.1 RNA PROCESSING

Much of a gene's activity is controlled at the transcriptional level through a plethora of DNA regulatory elements, such as Transcription Factors (TFs), which regulate the expression of a gene as a response to internal and external stimuli, for instance through modulation of chromatin accessibility [22]. However, unlike in prokaryotes, transcription of DNA to RNA and translation of RNA to proteins occurs within different cellular compartments, which are separated by the nuclear membrane [23]. On its way from the sites of transcription to compartments of translation, precursor messenger RNA (pre-mRNA) is subject to extensive processing, such as chemical modification, splicing, export and degradation, which contribute substantially towards the abundance and functional properties of the resulting gene product. The sequence of pre-mRNAs can be structured into distinct functional compartments, owing to their role in RNA processing. During splicing (Section 2.1), non-coding *introns* are removed from the pre-mRNA sequence, while *exons* are retained. At the 3′ and 5′ ends, mRNAs contain UTR sequences that can be hundreds of nucleotides in length and which are densely populated with RNA regulatory motifs that govern processes such as RNA localization (Section 2.1) and RNA degradation (Section 2.1). The anatomy of pre-mRNAs is illustrated in Figure 1.

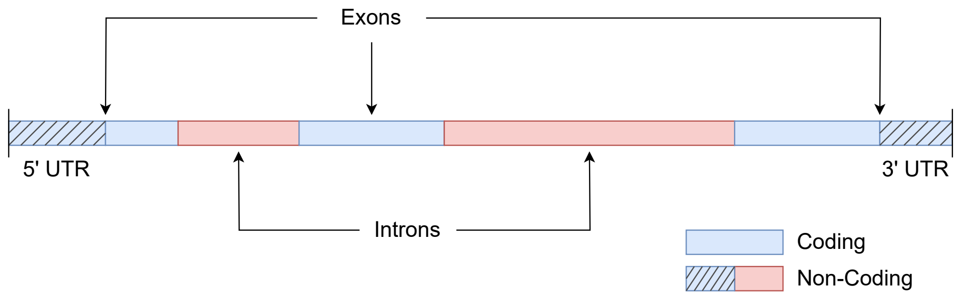


Figure 1: Organizational structure of pre-mRNA. Intronic regions are removed during post-transcriptional RNA processing (Section 2.1), while coding exonic regions as well as UTR sequences at the 5′ and 3′ ends of the transcript remain.

5′-capping and (alternative) 3′ poly-adenylation

Shortly after the initiation of transcription by polymerase II, the 5′ end of the nascent transcripts is co-transcriptionally capped with a N7-methylated guanosine in a process that is referred to as *5′-capping* [24, 25]. Capping of mRNA is highly conserved in eukaryotes and plays an important role in the initiation of protein synthesis as well as protection of the nascent transcript from premature degradation via 5′ to 3′ exonuclease cleavage [24]. Following 5′-capping, the 3′ end of the nascent RNA is cleaved and subsequently polyadenylated via interaction of the Cleavage and Polyadenylation (CPA) machinery with the Poly-Adenylation Signal (PAS), which is usually contained in the last exon of the pre-mRNA transcript [26]. However, Alternative Polyadenylation (APA) via use of alternative PAS is a widespread phenomena in higher eukaryotes, which contributes significantly towards transcript diversity. For instance, while use of alternative PAS in the last exon results in mRNA isoforms that code for the same protein, it generates transcripts with truncated or elongated 3′ UTR sequences, which may contain a number of regulatory elements that control mRNA and protein localization or RNA stability [26]. Further, intronic PAS can result in skipping of the last exon, thereby directly affecting the protein sequence. Aberrant CPA, for instance as a consequence of mutations in the genes coding for CPA-associated factors or PAS of individual genes has been shown to be associated with human diseases, such as haematological disorders and muscular dystrophy [27].

(Alternative) RNA Splicing

In eukaryotes, pre-mRNA transcripts are further subject to processing via splicing, where *intronic* regions that interspace *exons* are removed, such that the resulting transcript is composed of coding regions and UTRs. In higher eukaryotes, splicing of a single pre-mRNA often results in a number of different mRNA isoforms through alternative splicing, a processes that contributes significantly towards proteome diversity. In humans, it is estimated that over 90% of the transcriptome is spliced alternatively [28]. Alternative splicing may generate distinct RNA isoforms in several ways, including use of alternative 3′ and 5′ splice sites, mutually exclusive exons or intron retention [29]. Aberrant splicing is associated with numerous human diseases and genetic disorders [30], including tumor initiation and progression [29]. Here, modulation of isoform distributions, for instance through splice site-switching antisense oligonucleotides [31], is a promising target for therapeutic applications.

RNA Export and Localization

Following transcription and processing, mRNAs need to be co-localized with ribosomes for translation to proteins. In eukaryotes, this involves export of the mRNAs from the nucleus to the cytoplasm. It has been shown that the distribution of mRNA across cellular compartments is crucial for maintaining homeostasis and is governed by an intricate RNA localization system, which modulates local RNA concentration [5, 32]. For instance in neurons, the axon may extend centimeters away from the cell's nucleus, such that the localized synthesis of proteins is required in order to react to external or internal stimuli [33]. Consequently, specific mRNAs need to be transported and stored in distant locations in order to be readily available for translation. Several RNA localization mechanisms exist, including diffusion and directed transport via motor proteins along microtubules. Alternatively, mRNAs may "hitch-hike" on other cellular organelles, such as lysosomes or endosomes, for long-distance transport [5, 34, 35]. On the transcript level, RNA transport is controlled via localization signals, short sequence elements that are predominantly located in the 3′ UTRs of mRNAs and recognized by cis-acting localization factors [5].

RNA Degradation

A key role of the RNA degradation machinery is the surveillance and quality control of transcripts, to prevent the further (potentially energy-intensive) processing of faulty RNAs as well as to avoid the potentially deleterious effect of misfolded proteins. Transcript-monitoring pathways include the nonsense-mediated decay machinery, which facilitates degradation of truncated RNAs with a premature stop codon [36, 37]. Several mechanisms of cellular RNA-degradation exist, included hydrolyzation of RNA from the 5′ or 3′ ends via exonucleases or internal fragmentation via endonucleases [38]. Since cellular mRNA levels are jointly controlled by the rate of transcription of nascent RNA as well as the degradation of existing RNAs in the cytoplasm, regulation of mRNA decay is an important additional mechanism to determine the rate of protein production [36]. For instance, processes such as cell fate decisions and the transition between cell phases during the cell cycle have been shown to be regulated on the level of mRNA half-life through targeted degradation of RNAs [37].

RNA Modifications

Transcripts are subject to over 100 distinct chemical modifications, with methylation via N6-methyladenosine (m6A) being the most prevalent one [39]. Numerous studies have elucidated the extensive role of m6A in the RNA metabolism, which acts through an intricate system of, m6a writer-, reader-and eraser-proteins [40, 41]. RNA modification with m6a has been implicated in translation dynamics, where it may affect codon/anti-codon stability [39]. For instance, Mao et al. [42] showed that m6A in mRNA Coding Sequences (CDSs) promotes translation through YTHDC2, a m6A reader protein. Further, m6a methylation is involved in APA, where isoforms with m6a may favor proximal APA sites and thus short 3′ UTRs [39], as well as the nuclear export of mRNAs. Misregulation of the m6a modification pathway has been shown to be associated with various types or cancer and other human diseases [43].

2.2 RNA-BINDING PROTEINS (RBPS)

RBPs are a large class of proteins that are involved in every aspect of post-transcriptional RNA processing and regulation. While previous studies curated at set of ~1,500 RBPs present in human genes [44], recent data from RNA Interactome Capture (RIC) studies (Section 3.1) suggests that well over 2,000 RBPs exist, rendering it one of the largest protein families [45]. In addition, RBPs are highly conserved, more so than transcription factors [45], highlighting their crucial role in controlling cellular processes. However, many RBPs were shown to exhibit DNA-binding capabilities and engage in Protein-Protein-Interactions (PPIs) [46, 47]. RBPs identify and bind their RNA targets via recognition of short sequence or structural motifs and form Ribonucleoprotein Particles (RNPs). Binding is usually facilitated through one or more RNA-binding Domains (RBDs), which may have specific sequence or structural affinities, with combinations of several domains giving rise to highly specific binding footprints [47]. Although many RBD have been characterized [48], their distribution among RBPs is not uniform. RNA Recognition Motifs (RRMs), which are on average 90 amino acids in size and interact with 2–8nt long single-stranded RNA (ssRNA) [47], are by far the most numerous and are present in at least 200 human RBPs [44], with estimates suggesting that up to 1% of human proteins harbor RRM domains [49]. Other domains include the 70 amino acid K Homology (KH) domain, which recognizes 4nt short ssRNA GXXG motifs, the Zinc Finger (ZF) domain, with subtypes such as zinc knuckles recognizing RNA stem-loop structures and the Pumilio Homology (PUM) domain or PUF, which recognizes short ssRNA with low affinity [47]. Beyond RBPs with defined domains, a large number of "unconventional" RBPs that lack canonical RBDs exist, with estimates suggesting that up to half of all human RBPs may bind through unknown or unconventional mechanisms [46, 50, 51]. Many of these proteins where shown to binding RNA through their Intrinsically Disordered Regions (IDRs) [51], regions of a proteins that do not fold into stable or well-defined structures. These RBPs have been implicated in liquid-liquid phase separation and the formation of RNP granules, such as stress-granules [46].

RBPs play a critically role in all aforementioned RNA processing pathways (Section 2). For instance, CPA specificity factor (CPSF) binds the hexamer PAS motif AAUAAA and, together with other factors such as CFI and CFII, initiates cleavage and subsequent polyadenylation [26]. In RNA modification such as m6a, methylation is facilitated by RNA-binding writer proteins, such as methyltransferase-like 3 (METTL3) and METTL14, while m6a modifications may be removed by eraser proteins such as fat mass and obesity-associated protein (FTO) and alkB homologue 5 (ALKBH5) [39]. Reader proteins specifically bind to m6a modified RNA, including YTHDF1 which promotes translation via the recruitment of translation initiation factors [39]. In context of splicing, proteins of the SR family, such as U2AF, directly interact with the pre-mRNA by binding to the 5′ and 3′ splice sites and promote the inclusion of exons [52]. Indeed, splicing-associated Single-nucleotide Variants (SNVs) are associated with the disruption of binding sites of splicing-associated RBPs, such as SRSF1 or SF3A3 [3]. Lastly, in the context of RNA stability, some RBPs may facilitate degradation of RNA through binding to AU-rich Elements (AREs) and subsequent recruitment of deadenylases, or stabilize transcripts via competitive binding by influencing the accessibility of RNA to endo-and exonucleases [36, 37].

RNA-binding Proteins as Host-Factors in Viral Infection

In addition to maintaining cellular homeostasis, RBPs play a critical role in the context of viral infection. Viral RNAs are bound extensively by human proteins, indicating that viruses exploit the host cell's RNA processing pathways [7, 53, 54]. Schmidt et al. [7] recently utilized RAP-MS (Section 3.1) to identify 104 human proteins directly binding to SARS-CoV-2 RNAs. Moreover, they demonstrated that perturbing the RBPs CNBP and LARP1 results in a significant reduction in viral replication. This suggests that host RBPs may be crucial for establishing a permissive host environment for viral infection and sustained replication. At the same time, RBPs are involved in detecting and subsequently degrading viral RNAs, thereby playing a vital role in antiviral defense [55]. Flynn et al. [9] identified 309 human RBPs interacting with SARS-CoV-2 via ChIRP-MS and showed that the majority of these proteins have antiviral activity. Together, this underscores the important and diverse roles of RBPs in viral infection [8].

3 MEASURING PROTEIN-RNA INTERACTION

As pathways of post-transcriptional regulation are largely governed by RBPs, identifying RBPs and mapping protein-RNA interactions proteome-and transcriptome-wide is a key challenge in RNA biology research. To this end, several experimental methods have been developed, which can broadly be categorized into RNA-centric and protein-centric approaches [56, 57].

3.1 RNA-CENTRIC METHODS

RNA-centric methods aim to identify the set of proteins bound to a specific RNA of interest. McHugh et al. [58] modified RNA Antisense Purification (RAP) to isolate a targeted RNA and measure bound proteins via Mass Spectrometry (MS) in vivo. Their protocol, RAP-MS, first creates covalent bonds between RNA and proteins engaged in direct interaction via irradiation with Ultraviolet (UV) light. The RNA of interest, together with covalently bound proteins, is then purified via anti-sense DNA probes complementary to parts of the target RNA and bound proteins are subsequently identified via mass spectrometry. To broadly identify RBPs that bind to mRNAs *in vivo*, RIC methods have been developed [59, 60], which crosslink RNAs and bound proteins, followed by selection of polyadenylated RNAs via oligo(dT) magnetic beads. Improvements of the RIC protocol have since been proposed, such as Enhanced RIC (eRIC) [61], which improves specificity via modification of the capture probes. RIC-based studies have been largely responsible for uncovering the prevalence of RBPs across various species. For instance, Hentze et al. [62] maintain the RBPbase database, which aggregates RIC experiments for eukaryotes such as human, mouse, drosophila and yeast. To study binding of RBPs to non-coding RNAs, ChIRP-MS was developed by Chu et al. [63]. Similar to RAP, ChIRP-MS utilizes covalent cross-linking of proteins to RNAs, followed by capturing of target RNAs via anti-sense oligonucleotide probes and identification of bound proteins via MS, however ChIRP-MS performs cross-linking via treatment with formaldehyde. Notably, Labeau et al. [64] utilize the ChIRP-MS protocol to identify human RBPs that interact with the SARS-CoV-2 RNA.

While RNA-centric methods can uncover the protein interactome of a given RNA of interest, they only identify if — not where — a protein is bound to the RNA. Therefore, RNA-centric methods identify protein-RNA interactions at low spatial resolution, which may be insufficient for the exact characterization of RBP-binding motifs. Further, to identify the interactomes for several transcripts, dedicated experiments have to be performed on a transcript-to-transcript basis, which is prohibitive for the study of protein-RNA interaction at a transcriptome-scale. To enable high-resolution, transcriptome-wide identification of binding sites for a RBP of interest, protein-centric methods have been developed.

3.2 PROTEIN-CENTRIC METHODS: CLIP-BASED APPROACHES

Protein-centric methods generally aim to identify the RNA targets of a protein of interest. Several *in vitro* approaches for studying RBP-binding have been developed, including RNAcompete [65], RNA Bind-n-seq (RBNS) [66] and RNA-SELEX [67]. These protocols measure the affinity of an RBP to a pool of random RNA fragments. One of the most prevalent protein-centric approaches for studying *in vivo* RBP-binding transcriptome-wide is Crosslinking-Immunoprecipitation (CLIP) followed by Sequencing [68] and its derivatives [12]. These methods generally rely on the generation of covalent crosslinks between proteins and bound RNAs, which is followed by purification of protein-RNA complexes via Immunoprecipitation (IP) using protein-specific antibodies, reverse-transcription of RNA fragments and finally sequencing [12]. Several variants of CLIP have been developed, including protocols that enable detection of protein-RNA interactions at nucleotide resolution, such as Photoactivatable Ribonucleoside-enhanced CLIP (PAR-CLIP) [69], Individual-nucleotide CLIP (iCLIP) [70] and Enhanced CLIP (eCLIP) [14]. As eCLIP is the primary CLIP protocol used in the ENCODE [71] database (Section 3.2.4), the data from which was used in all three core publications [1–3], its steps are described in detail below. A comprehensive overview of CLIP-based methods for profiling protein-RNA interactions is given by Hafner et al. [12].

3.2.1 *Enhanced CLIP (eCLIP)*

Enhanced CLIP allows for the detection of protein-RNA crosslinks at single-nucleotide resolution by relying on truncations generated during reverse-transcription as the primary diagnostic event, similar to iCLIP [13]. An outline of the eCLIP experiment workflow is depicted in Figure 2. In the initial step of eCLIP, covalent cross-links between proteins and their bound RNAs are generated by irradiating cells with UV light (Step 1). Subsequently, cells are lysed and RNA is fragmented via RNase I [14] (Figure 2, Step 2). Next, 98% of the lysate is immunoprecipitated using a protein-specific antibody to isolate RNA fragments which are bound to the target protein, while the remaining 2% is used for the generation of a Size-match Input (SMI), which omits the IP step and is subsequently used as a control (Steps 3.1 and 3.2). Following 3′ adapter ligation (Step 4), IP and SMI samples are applied onto a gel (nitrocellulose membrane) and protein-RNA complexes of both samples are size-selected at up to 75kDa over the target protein, to yield RNA fragments of the appropriate size (Step 5). Note that this size-selection step may result in low-to-moderate enrichment of protein-RNA complexes of the target protein in the SMI samples, depending on the abundance of the protein [72]. Bound proteins are then digested by proteinase K (Step 6), which is followed by reverse-transcription of RNA fragments to Complementary DNA (cDNA) (Step 7). Digestion by proteinase K commonly leaves a covalently bound peptide at the crosslinked site, leading to a premature stop of the reverse-transcriptases. As a result, only a fraction of cDNA fragments represent a full-length read-through, while the majority of cDNA sequences are truncated at the crosslinked site. These truncation events are then used as a diagnostic signal for the detection of protein-RNA interactions at nucleotide-resolution in eCLIP and iCLIP. After extraction of cDNA via RNA removal (Step 8) and 3′ adapter ligation (Step 9), cDNA is amplified via Polymerase Chain Reaction (PCR) (Step 10), which is followed by paired-end high-throughput sequencing (Step 11). Here, the start-positions of the resulting R2 reads correspond to the cDNA truncation sites. As a final step, reads are aligned to the reference genome (Step 12) and the eCLIP crosslink-count signal is computed by building a pile-up of R2 read start positions (Step 13). This signal may then serve as input to *peak callers*, in order to identify crosslink sites that are significantly enriched over the control experiment and thus likely represent genuine sites of protein-RNA interaction (Section 3.2.3).

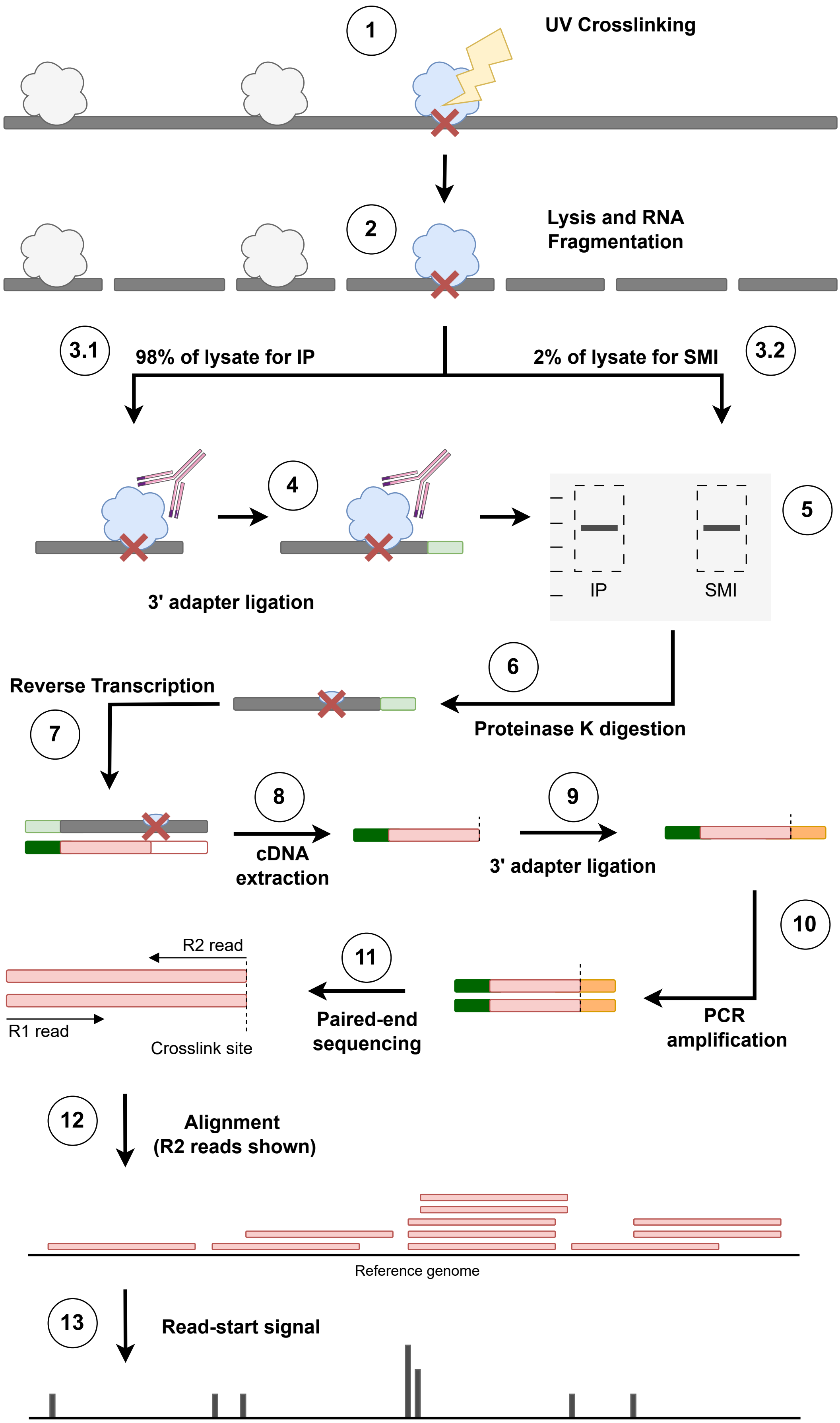


Figure 2: Schematic overview over the steps involved in the eCLIP protocol. Parts of this figure were adapted from Van Nostrand et al. [14].

3.2.2 *Sources of CLIP Bias and Noise*

While CLIP allows for the accurate and high-resolution profiling of protein-RNA interactions at a transcriptome-wide scale, it is subject to various sources of variability and bias, which complicate downstream bioinformatics analysis of CLIP data [12]. In the following, an overview of CLIP biases and noise is given.

- **Competitive-and co-binding:** Several RBPs have been shown to co-bind at specific RNA locations, for instance through the formation of heterodimers or targeted recruitment of co-binding factors through PPIs [12]. Thus, the CLIP experimental signal of a target RBP may be enriched with binding of one or more synergistically binding RBPs. On the other hand, RBPs may be in competition such that binding of the target RBP to a specific site is prevented by the occupancy of another RBP. In this case, binding of the target RBP at those sites may not be detected in CLIP, which is especially the case if antagonistic RBPs are expressed at higher levels.
- **Variability in binding strength:** Different RBPs exhibit different binding affinities, which may be subject to sequence and structural features of the RNA interaction site. Generally, weak binding affinity may lead to short-lived, transient interactions [12], while a stronger binding affinity leads to a longer occupancy of binding sites. As CLIP only represents a snapshot of binding at a specific point in time, binding of weakly binding RBPs may not be detected at many sites.
- **Co-purification with other RBPs:** During CLIP, the IP step results in the purification of RNA fragments which are covalently attached to the protein of interest through a protein-specific antibody. Affinity of antibodies to their target protein may vary, which can lead to contamination of the CLIP library with other RBPs and/or inefficient purification. While protocols such as iCLIP and PAR-CLIP generally perform a quality control step to increase specificity via visualization of pulled-down protein-RNA complexes with SDS-PAGE [12, 73], this step is omitted in eCLIP. Therefore, eCLIP may be particularly prone to contamination with other RBPs. Indeed, recent studies suggest that contamination is present at higher levels in eCLIP, compared to iCLIP and PAR-CLIP, with several eCLIP experiments showing an enrichment of G-rich motifs around crosslink sites, which are hypothesized to stem from co-purified RBPs [73, 74]. This highlights the importance of a SMI control for eCLIP, although it has been suggested that signal of the target protein itself may be enriched in the control, thereby complicating the detection of genuine sites of protein-RNA interaction [3, 12].
- **Transcript/RBP abundance:** CLIP experiments are subject to protein and RNA abundances in the cell type at the time of the experiment. The detection RBP-binding may therefore be precluded by the low expression of RNAs as well as dominating signal from a few highly expressed RNAs. Further, a limitation of CLIP is that no claim on the relative affinity of RBP-binding between two binding sites can be made if those sites are located on different transcripts, as differences in observed CLIP signal may merely reflect the difference in expression of those transcripts [72].
- **Crosslinking-related biases:** Crosslinking biases may be introduced directly through UV irradiation, a step that is shared across many CLIP protocols. For instance

> Wheeler et al. [72] suggest that pyrimidine nucleotides are more photoreactive than purines and therefore crosslink more easily. Haberman et al. [75] show that when performing motif analysis on SMI data of eCLIP, motifs of all experiments are rich in uridines, indicating that UV preferentially creates crosslinks at uridines. Further, Sugimoto et al. [76] find that CLIP and iCLIP results in a strong and mild uridine enrichment, respectively, while this enrichment is absent when using non-UV crosslinking methods. Together, these studies strongly suggest that UV irradiation introduces cross-linking biases in a sequences-specific manner. Besides nucleotide-biases, different amino acids show a variable crosslinking reactivity, with a particularly high reactivity for amino acids such as Cysteine, Tyrosine and Phenylalanine [72]. Thus, contamination of samples with RBPs that exhibit strong crosslinking efficiency may particularly obscure the genuine binding of the target RBP. Wheeler et al. [72] further suggest that crosslinking efficiency towards double-stranded RNA may be low, due to the inability of amino acids to access the paired nucleotides. This biases CLIP towards RBPs that preferentially bind to single-stranded.

Considering the numerous biases in CLIP, machine learning methods must employ strategies to avoid including biased features in their decision-making processes. In [1], Pysster models are trained by generating negative-class instances for a target RBP by sampling positives from other RBPs, thereby ensuring that CLIP biases are incorporated in the positive and negative sets at equal rates. A similar strategy is applied in [2], in order to benchmark RBP-binding prediction methods in an unbiased manner. In [3], RBPNet explicitly models the distribution of crosslink counts for the SMI control experiment, allowing it to later separate bias from protein-specific signal.

3.2.3 *Peak Calling*

Following the alignment of CLIP reads to the reference genome, a key post-processing step is the identification of regions which show elevated levels of CLIP signal, a process which is commonly referred to as "peak calling". As this process discretizes transcriptome regions into bound and unbound regions, it is pivotal for the training of machine learning classifiers for protein-RNA interaction prediction. Several peak callers have been developed in recent years, including CLIPer [77, 78] (an in-house peak caller of the ENCODE [71] database), PARalyzer [79], Piranha [80] and Clippy [81]. These methods either utilize the local read coverage or protocol-specific signal footprints, such as truncation or T-to-C conversion counts, for iCLIP and eCLIP, or PAR-CLIP, respectively [82]. Recently, Krakau et al. [83] developed PureCLIP, a single-nucleotide peak caller for eCLIP and iCLIP data which utilizes a Hidden Markov model.

3.2.4 *Public CLIP Datasets*

Post-transcriptional processes are commonly governed by dozens of RBPs. At the same time, many RBPs are involved in several post-transcriptional pathways [74]. To study post-transcriptional processes in their entirety, profiling protein-RNA interaction of a large number of RBPs is required. However, while progress towards multiplexing CLIP experiments has been made [84, 85], current CLIP protocols require that experiments are performed on a protein-by-protein basis, such that the large-scale CLIP studies are associated with significant time and monetary costs. Therefore, publicly available CLIP databases, which are usually generated through larger consortia or collected across multiple individual studies, are an important resource for the study of the human protein-RNA interactome. Several datasets exists, including doRiNA [86, 87], iONMF [88] and ENCODE [14, 71], some of which have been reviewer by Horlacher et al. [2]. Most notably, the Encyclopedia of RNA elements (ENCORE), maintained by the ENCODE consortium [71], hosts datasets of over 200 eCLIP experiments, across more than 150 RBPs, generated by Nostrand et al. [14]. As such, it represents the largest publicly available collection of CLIP datasets. A distinguishing feature of the ENCODE database is that all eCLIP experiments were conducted in the same research group and pre-processed with the same computational pipeline, with data being available in the form of processed protein interactions (*BED* format), control-normalized signal (*bigWig* format), alignment files of processed reads (*BAM* format) and raw sequencing files (*FASTQ* format). Data from the ENCODE database was used in all three core publications supporting this thesis. For instance, RBPNet [3] models were trained on over 100 ENCODE eCLIP datasets of the HepG2 cell line, while Pysster and DeepRiPe models trained on ENCODE eCLIP data were used for the prediction of human-virus protein-RNA interaction [1]. In [2], several RBP-binding prediction methods were benchmarked using 223 eCLIP datasets from the ENCODE database. Since ENCODE profiles the RNA-binding of several RBPs in both HepG2 and K562 cell lines, this provided a unique opportunity to evaluate the generalization power of trained models across different cell types.

3.3 RBP-BINDING MOTIFS

Identifying binding motifs of RBPs is crucial for understanding how RBPs recognize their targets. Sequence specificities of DNA-or RNA-binding proteins are commonly represented in the form of Position-weight Matrices (PWMs) or Position-probability Matrices (PPMs), which describe the probability of finding a particular nucleotide at each position within a binding site. Early methods for prediction of protein-RNA interaction from RNA sequence utilized PWMs to scan for binding sites in arbitrary RNA sequences [89], while deep learning based approaches are more prevalent today [2]. PPMs and PWMs are constructed by first aligning sub-sequences at and around RBP-binding sites obtained from high-throughput experiments, such as CLIP (Section 3.2). Subsequently, nucleotide counts are aggregated into Position-frequency Matrices (PFMs), before normalizing the counts to PPMs and PWMs. An example is given in Figure 3. Note that the terms PWM and PPM are sometimes used interchangeably to describe a RNA or DNA motif.

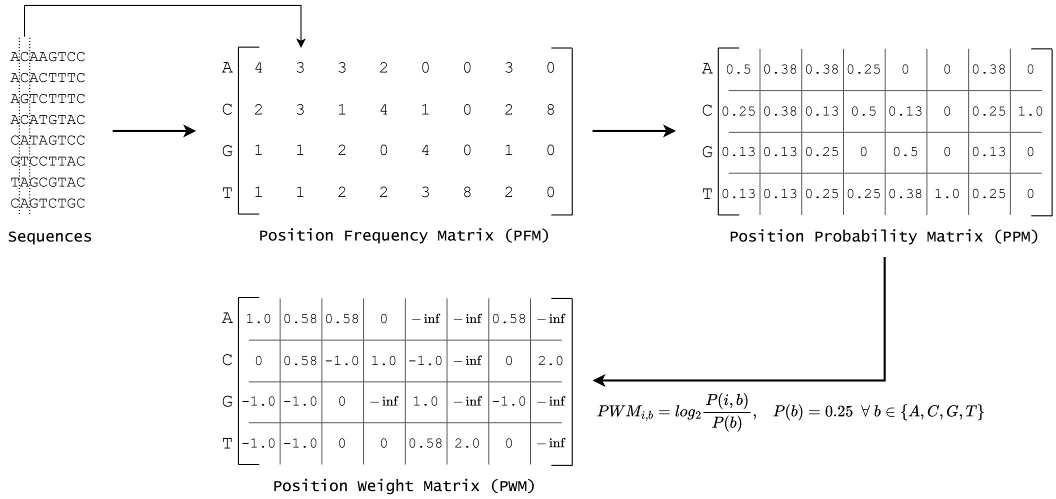


Figure 3: Representing binding motifs. Starting from a set of aligned sequences, a PFM is constructed by counting the number of occurrences of each base b at each position i. The PFM is then normalized to a PPM, before computing a PWM by converting the entries of the PPM to log-likelihoods using a background model of nucleotide probabilities $P(b)$, for instance a uniform distribution over the nucleotides A, C, G, T.

3.3.1 *RBP-binding Motif Databases*

Several databases exist which aggregate RBP-binding motifs. In the following, a brief description of a subset of popular RBP-binding motif databases is provided.

- *RBPDB* [90]: RBPDB aggregates binding data of 272 RBPs, gather from *in vivo* and *in vitro* experiments, and derives 71 primary sequence binding motifs either from literature (for *in vivo* data) or via manual construction of PWMs (for *in vitro* data). RBPDB also indicates whether a given RBP preferentially binding RNA secondary structures, although no structure motifs are provided.

- *ATtRACT* [91]: The ATtRACT database compiles 1,583 consensus RBP-binding motifs across 370 RBPs, by integrating motifs from other databases, including RBPDB [90] and CISBP-RNA [92]. The web-server allows for motif scanning in user-provided sequences as well as identification of de novo motifs.

- *mCross* [93]: mCross is a method to jointly model experimentally identified sites of protein-RNA interaction and RBP binding motifs. It was applied to 112 RBPs from the ENCODE [74] database and is accompanied by a database web-server for motif lookup.

- *oRNAment* [94]: The oRNAment database contains 454 motifs across 223 RBPs, spanning human as well as four additional model organisms. Notably, *in vitro* motifs were partially obtained from RNAcompete [65] experiments via the CISBP-RNA [92] database and constructed de novo from RNA-binding-N-seq (RBNS) [66] from the ENCODE [74] database.

- *RBPmap* [89]: The RBPmap database spans 165 motifs across 145 distinct RBPs for human, mouse and drosophila in MEME [95] format. It includes a web-server that allow mapping of protein-RNA interaction sites for a query RNA sequence via motif matching.

Part III

COMPUTATIONAL BACKGROUND

4 NEURAL NETWORKS

This chapter provides a brief introduction to neural networks. This thesis emphasizes Convolutional Neural Networks and techniques employed in the core publications, including *dilation* and *residual connections*. However, it provides limited coverage of other popular deep learning architectures like Long Short-Term Memory (LSTM) networks or Transformers. For a more comprehensive theoretical or practical understanding, readers interested in delving deeper can refer to Goodfellow et al. [96] or Chollet [97].

4.1 ARTIFICIAL NEURAL NETWORKS

Artificial neural networks, hereafter referred to as *neural networks* or simply *networks*, are a class of computational models that emerge through chaining of one or more (parameterized) functions. A neural network can generally be represented as a Directed Acyclic Graph (DAG) of operations which transform a set of input variables X to a set of output variables Y (Figure 4) [96].

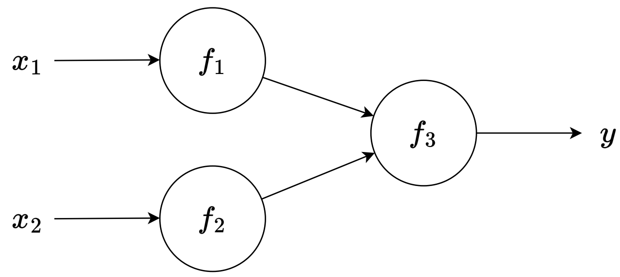


Figure 4: A schematic illustration of a neural network that chains three functions, f_1, f_2 and f_3. The depicted DAG is a visualization of the following expression: $f(x_1, x_2) = f_3(f_1(x_1), f_2(x_2))$. This example was adapted from Goodfellow et al. [96].

4.1.1 *Multi-Layer Perceptrons (MLPs)*

In the late 1950s, Frank Rosenblatt [98] presented the *perceptron*, a simple processing unit that, given a set of inputs $x_1, ...x_n$, computes a weighted average of the inputs using weights $w_1, ..., w_n$, which may be subsequently thresholded to perform a classification of the form $f : \mathbb{R}^n \to \{0, 1\}$. An example of Rosenblatt's perceptron is given in Figure 5. Note that a bias term may be added, which is equivalent to the intercept in linear regression. The (0-thresholded) perceptron function for input $\mathbf{x}$ may be written as

$$f(x) = \sigma(x^T w + b), \text{ where } \sigma(x) = \begin{cases} 1, & \text{if } x > 0 \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

A limitation of the perceptron is that its decision boundary is linear in the input space and can thus only be effectively applied to linearly separable problems. For instance,

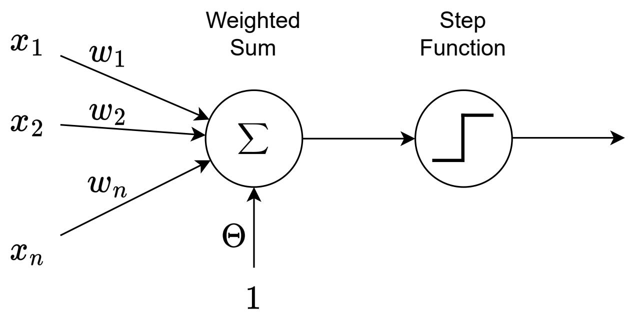


Figure 5: Depiction of Rosenblatt's perceptron. The perceptron unit takes as input a set of values $x_1, ..., x_n$ and computes a weighted average using weights $w_1, ..., w_n$. For classification tasks, the weighted sum may be thresholded via a step function, resulting in a $0-1$ binary classification. An additional bias term, which does not depend on the inputs, may be added to shift the decision boundary of the weighted sum of inputs.

while a perceptron may solve OR or AND problems, no single line in the input space can separate the two classes in the context of the XOR problem (Figure 6) [99].

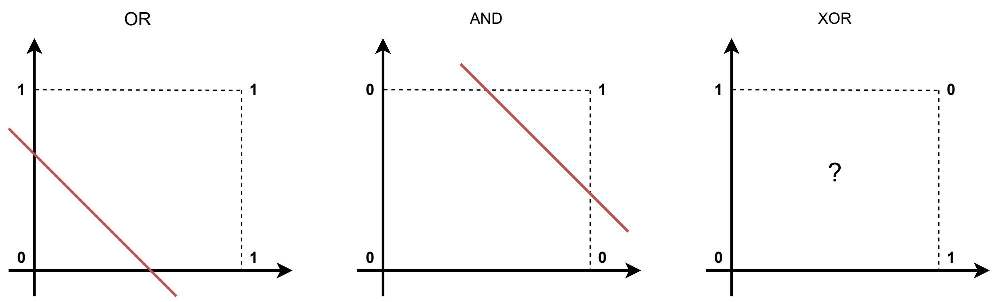


Figure 6: OR, AND and XOR problems. While linear decision boundaries exist for both OR and AND, the XOR problem can not be solved linearly. Parts of this figure were adapted from Rojas [99].

A solution is the addition of one or more *hidden* units via stacking of multiple perceptron *layers*. Consider the Multi-Layer Perceptron (MLP) given by in Figure 7, which can be written as

$$f_{MLP}(x) = \alpha(x^T \boldsymbol{w}^1 + b^1)^T \boldsymbol{w}^2 + b^2 \text{ [96]}, \tag{2}$$

where w^1, b^1 and w^2, b^2 are the weights and bias terms of the first and second network layers, respectively, and α is some non-linear *activation function*, for instance a *rectified linear unit* (Section 4.1.3). The first layer of the MLP projects the inputs into a new (hidden) space. By choosing the weights w^1 and bias term b^1 appropriately, samples of the XOR problem become linearly separable in the hidden space (Figure 7), which serves as input to a single output perceptron (Eq. 1) [96]. The use of a non-linearity in the hidden layer is crucial, as the chaining of two linear functions f and g would produce a new linear function [96].

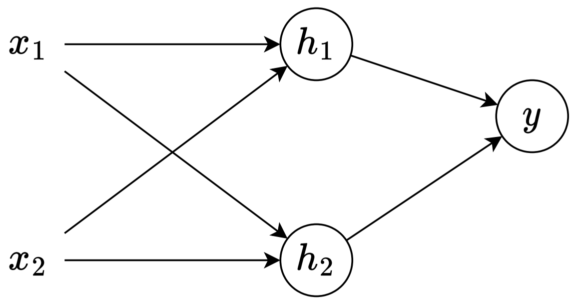


Figure 7: Illustration of a MLP with one hidden layer with hidden units h_1 and h_2.

For an in-depth example of solving the XOR problem with a MLP, readers may refer to Chapter 6.1 of Goodfellow et al. [96]. By chaining several layers with a sufficient number of hidden units, MLPs act as universal function approximators [100].

4.1.2 *Training Neural Networks*

Problems can rarely be solved by manually parametrizing a neural network. Instead, parameters are learned via a gradient-based optimization algorithm, such as Stochastic Gradient Descend (SGD). Gradient-based optimization algorithms compute the derivatives of the network parameters with respect to some *loss* function, which scores the error of the network on a set of training samples, and update the network parameters via gradient descent [101]. This process is typically carried out iteratively, wherein the network's parameters are repeatedly updated by computing the loss over subsets of the training dataset, known as *mini-batches*. Loss functions are task specific and the choice of an appropriate loss function is vital for the network to solve a given task. In the context of binary or multi-class classification, popular choices include binary or categorical cross-entropy, respectively. A detailed review of common task-specific loss function is omitted for brevity and the reader is referred to a recent survey on loss functions by Wang et al. [102].

Due to non-convexity of most loss functions, convergence to a globally optimal set of parameters is generally not guaranteed for gradient-based optimization algorithms [96]. After the introduction of vanilla SGD, a range of optimization algorithms have been devised, including Adagrad, RMSprop and Adam [103]. In the context of this thesis, both RBPNet [3] and Pysster [104] models were trained with the Adam [105] optimizer.

4.1.3 *Activation Functions*

Activation functions play a pivotal role in neural networks. Generally, a distinction must be made between activation functions of inner units and those of the network's last layer. While the main purpose of the former is to introduce non-linearity to the network and therefore is crucial for the network's ability to approximate arbitrary functions, the latter is task-specific and responsible for mapping the network's output to a suitable range of values. However, a common property of activation functions is that they have to be (at least piece-wise) differentiable.

A common choice for inner activation functions is the Rectified Linear Unit (ReLU), a piece-wise linear function of the form $f(x) = max(0, x)$. While early networks commonly made use of *sigmoid*[1] and *tanh*[2] activations, these functions are less common nowadays for inner unit activation, due to issues associated with vanishing gradients as a result of exceedingly high or low input values [106]. Although ReLU activation functions serve as a cornerstone in numerous state-of-the-art models and are widely considered the de facto standard, they are subject to issues such as the *dying ReLU problem* [107]. As a result, the search for new and improved activation functions is an active area of research and many successors to ReLU have been proposed in recent years. For instance, Hendrycks and Gimpel [108] introduced the Gaussian Error Linear Unit (GELU), which is smooth and, in contrast to ReLU, does not gate inputs by their sign. Ramachandran et al. [109] performed both an exhaustive and reinforcement learning-based search to systematically identify promising activation functions. Their best performing candidate, termed *Swish*, improved state-of-the-art performance on the ImageNet [110] benchmark and was shown to outperform ReLU across multiple domains, including computer vision and natural language processing. Misra [111] proposed *Mish* and identify a number of key properties of high-performing activations. First, activation functions should be unbounded above and bounded below, where being unbounded above avoids saturation effects, as is commonly observed for functions such as *sigmoid* and *tanh*, while being bounded below induces regularization. Further, functions should be continuously (rather than piece-wise) differentiable, to avoid singularities during gradient descent [111]. Lastly, non-monotonicity has been empirically shown to improve activation function performance and is a common property of Mish, Swish and GELU. In the following, the definitions of the ReLU, GELU, Swish and Mish functions are given.

- ReLU

$$f(x) = max(0, x) \tag{3}$$

- GELU

$$f(x) = x \times P(X \leqslant x), \text{ with } X \sim \mathcal{N}(0, 1) \tag{4}$$

- Swish

$$f(x) = x \times sigmoid(\beta x) \tag{5}$$

- Mish

$$f(x) = x \times tanh(softplus^3(x)) \tag{6}$$

1 $sigmoid(x) = \frac{1}{1+e^{-x}}$
2 $tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$
3 $softplus(x) = log(1 + e^x)$

For Pysster [104] and RBPNet [3] models trained over the course of this thesis, the ReLU activation function was used exclusively for inner network units.

Last-layer activation functions are chosen to transform the outputs of the last network layer to a range of values that is suitable for the task at hand. For instance, no activation function may be applied if the target values are unbounded below and above. In context of binary classification problems, the *sigmoid* function is a common choice for the output unit of the network, as it takes on values in the range of $(0, 1)$ and can therefore be interpreted as the probability of the foreground class. For multi-class classification problem, a *softmax* function is commonly applied, to establish a categorical distribution across classes. In BPNet [112] and its derivatives, such as RBPNet [3] and ChromBPNet [113], a last-layer *softmax* activation is used to parameterize a multinomial distribution of crosslink-or read counts over a DNA or RNA sequence.

4.2 CONVOLUTIONAL NEURAL NETWORKS

Convolutional Neural Networks (CNNs) are a class of networks that leverage the fact that for many tasks, natural spatial relationships exist between input features. While CNNs were first developed in the context of computer vision and the design of their architecture was initially motivated by the signal processing of the visual cortex [101], they have since been successfully applied to various sequence-based task in computational biology, such as the prediction of TF-and RBP-binding [114], as well as the prediction of open chromatin regions [4, 115].

At the heart of CNNs is the convolution operation, which can be formally defined as

$$(F * k)(p) = \sum_{s+t=p} F(s)k(t) \quad [116]. \tag{7}$$

Given a sequence F, a convolutional filter (also referred to as kernel) k is applied to each position p of F by systematically invoking it over local patches of the sequence, thus generating an output feature map. At each position p, the weights of the kernel are multiplied with the local patch and summed up to produce a scalar output value. An example of a 1D convolution operation applied to a RNA sequence is given in Figure 8.
Intuitively, convolution filters act as local feature extractors. For instance, they may detect the presence of horizontal or vertical edges (in the context of image recognition) or RNA sequence motifs (in context of protein-RNA interaction prediction). Stacking of multiple convolutional kernels enables the detection of high-level features through composition of lower-level features. By applying several convolutional kernels at each layer, informative high-dimensional feature maps are created. Neural networks that make use of convolutional layers as their primary mode of feature extraction are collectively referred to as CNNs. Often, the outputs of the last convolution layer of CNNs serve as inputs to a MLP (Section 4.1.1), which performs the final classification or regression

4 One-hot encoding is a binary representation of categorical variables where each category is represented by a distinct binary vector. For instance, a RNA base $b \in \{A, C, G, U\}$ may be represented by the vectors $A = [1, 0, 0, 0]$, $C = [0, 1, 0, 0]$, $G = [0, 0, 1, 0]$, or $U = [0, 0, 0, 1]$.

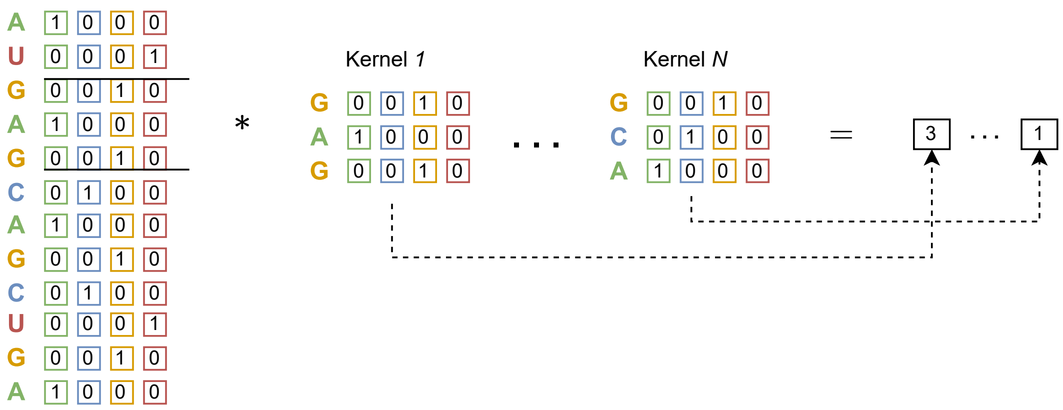


Figure 8: Example of a 1-dimensional convolution operation of size 3 is applied to an RNA sequence. A one-hot[4] encoded RNA sequence of length N is convolved by kernels $1, ..., N$. Kernels that match a given sequence patch well will generate a high kernel value (e.g. kernel 1 matching the GAG motif), thereby acting as local motif detectors.

task. However, purely convolutional networks may be used for sequence-to-sequence tasks, as in the case for RBPNet [3], which maps an RNA sequence to a distribution of crosslink counts over each nucleotide.

A key advantage of convolutions over densely connected MLPs is their parameter efficiency. As the same filters are convolved over the entire input, each layer requires only a fraction of learnable parameters. Further, convolutions are *translational invariant*, that is they can detect features within different parts of the image and are not sensitive to input translations such as shifts. In contrast, the number of parameters at each unit of an MLP layer is proportional to the number of input features and detection of features needs to be learned separately for each part of the input, resulting in statistical and computational inefficiencies.

4.2.1 *Stride, Padding and Pooling*

Applying a convolutional filter of size k over an input of length n results in an output feature map of size $n - k + 1$. To prevent the progressive reduction of the spatial size of the feature map when applying convolution operations repeatedly, *padding* can be applied to the input. For example, *same* padding dynamically adds padding values to the borders of the input to ensure that the output feature map has the same shape as the original input. This is crucial for tasks such as image segmentation or sequence-to-sequence applications, including BPNet and RBPNet [3, 112]. Additionally, the coarseness of the output feature map can be controlled using the *stride* parameter, which determines how the filter moves over the input. Smaller strides result in higher-resolution feature maps. Lastly, to increase the receptive field of convolutional filters in subsequent layers, feature maps are often down-sampled using *pooling* operations. These operations take a set of neighboring values as input and produce a summary statistic, such as the maximum or mean value. Stride, padding and pooling are demonstrated in Figure 9.

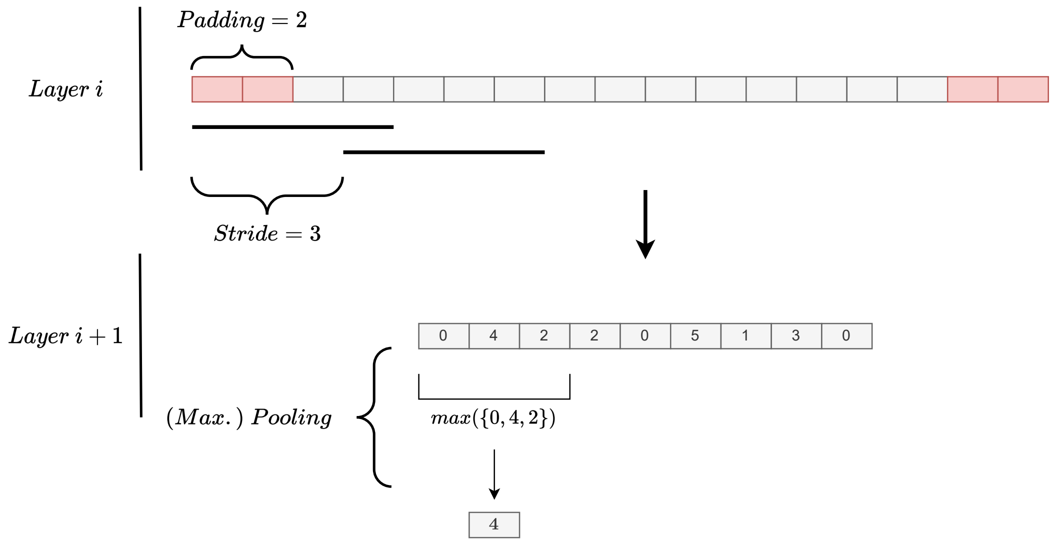


Figure 9: Illustration of *padding*, *pooling* and *strides* in context of CNNs.

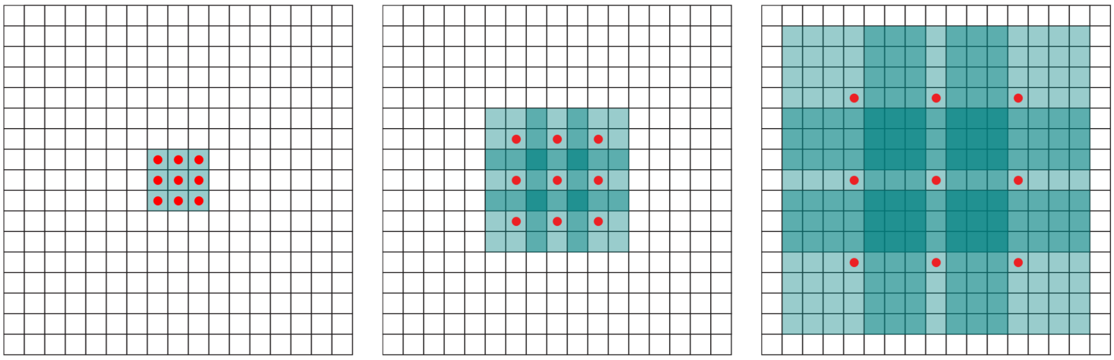
4.2.2 *Dilated Convolutions*

CNNs can detect complex features by learning feature hierarchies across layers. However, detection of a higher-level feature by a kernel k^l in layer l that spans a large portion of the input, such as a 2D object in the context of computer vision, requires that the object is in the *receptive field* of k^l. While the receptive field of a convolutional filter in a hidden layer of a CNN may be increased through pooling or by convolving the input with larger strides [117], many sequence-to-sequence or image-to-image tasks necessitate the preservation of the input resolution. This requirement precludes the use of spatial down-sampling of CNN feature maps. To address this issue, Yu et al. [116] propose the use of dilated convolutions. Formally, the definition of the convolution operation in Eq. 7 may be generalized by introducing a dilation factor d as

$$(F *_d k)(p) = \sum_{s+dt=p} F(s)k(t) \tag{8}$$

That is, the convolutional filter is applied with spatial gaps, allowing for an increased receptive field while maintaining a constant output size. This is demonstrated in Figure 10, which shows the receptive fields of convolutions with dilation factors 1, 2 and 4. By doubling the dilation factor in each successive layer, the receptive field is increasing exponentially while the number of parameters grows linearly (as in vanilla convolutions with $d = 1$), with no loss of resolution [116].

BPNet [112] and RBPNet [3] make use of 1D dilated convolutions to rapidly increase the receptive field of the network in order to learn potential cooperativity of distant motifs. Recently, dilated convolutions have been used successfully in place of transformers for the training of a large nucleotide language model [118].



(a) (b) (c)

Figure 10: Example of a 3×3 convolution operation with dilation factors 1 (**a**), 2 (**b**) and 4 (**c**), respectively. This figure was taken from Yu et al. [116] (Figure 1) with permission from the authors.

4.3 MODEL REGULARIZATION

Neural networks are prone to overfitting due to their ability to fit arbitrary functions exceptionally well. Consequently, in a limited data regime and without proper regularization, neural networks with sufficient capacity will fit the oddities of the dataset during training time, leading to low generalization performance. Formally, the error of a predictive model may be decomposed into three terms. The bias, which is a result of incorrect assumptions of the model, such as assuming linearity for a non-linear problem. The variance, which results from a model learning the non-informative noise of a dataset and the irreducible error, which represents the inherent randomness of the problem. Consequently, to improve model performance, one must either reduce the bias or variance of the estimator, which usually incurs a tradeoff, commonly referred to as the *bias-variance-tradeoff*. As neural networks are at the extreme ends of low bias, high variance models, further reducing bias of the network is usually of little concern. Instead, this section introduces several common strategies to reduce the network's variance in order to combat overfitting or improve convergence during training.

4.3.1 *Dropout*

A common strategy to reduce the variance of a predictor is via training of a model ensemble, for instance via bagging. Here, multiple instance of the same model are trained on boostrap samples of the dataset and predictions across all bootstrap models are averaged[5] at inference time. Examples for model bagging include tree-bagging and *random forests* [119].

As the computational burden associated with training neural networks often prohibits the creation of model ensembles, Srivastava et al. [120] developed *dropout*. The key idea of dropout is to randomly remove, or zero-out, a portion of units in each layer

5 In case each model is trained on a set of samples that is *i.i.d.* from all other datasets, the variance of the model can be reduced by σ^2/n, where n is the number of models in the ensemble. However, as bootstrap datasets and thus bootstrap models are correlated, the variance is instead reduced by $\rho\sigma^2 + \frac{1-\rho}{n}\sigma^2$, where ρ is the pairwise correlation [119].

of the network during training time, such that each forward pass is performed over a different *sample* of the network (Figure 11). At inference time, no dropout is performed such that the resulting prediction represents an average[6] over sampled networks. This regularizes the network by breaking down complex co-adaptions of network units that may be learned on the training dataset into simpler co-adaptions, as a given unit can not rely on the inputs from all preceding units [120].

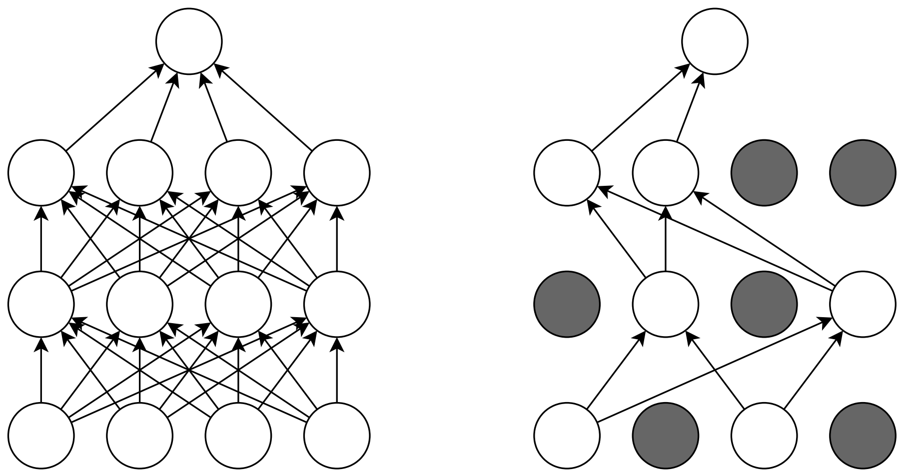


Figure 11: Example of neural network regularization via dropout. Shown is a vanilla neural network without dropout (**a**) and the corresponding version with dropout (**b**).

4.3.2 *Batch Normalization*

Ioffe and Szegedy [121] observed that as network parameters are updated during training, the distribution of each layer's inputs shifts, resulting in slow training as layers need to continuously adapt to shifts in their input. To reduce this *internal covariate shift*, they propose a normalization step based on mini-batch statistics during training. Formally, this *batch-normalization* step is defined as

$$\hat{x} = \frac{x^{(k)} - E[x^{(k)}]}{\sqrt{Var[x^{(k)}]}}. \tag{9}$$

That is, each unit is scaled to have zero mean and unit variance. To apply the batch-normalization transformation at inference time (where batches are usually not available), a population mean and variance are computed at train time, which are then used for normalization at test time. Batch normalization permits significantly higher learning rates, leading to faster convergence and increased robustness towards different network weight initialization strategies [121].

6 Note that at inference time, network weights are scaled with $W_{test} = pW$, where p is the unit retention probability.

4.4 RESIDUAL CONNECTIONS

Deep neural networks have long been challenging to train due to vanishing or exploding gradients [122, 123]. Interestingly, rather than leading to overfitting, these effects may lead to severe under-fitting, as network depth is increased [122]. To address these issues, He et al. [122] introduce the concept of *residual* or *skip* connections, which are depicted in Figure 12 and formally defined as $y = F(x) + x$. where x and y are input and output of a network layer and F represents the layer's mapping.

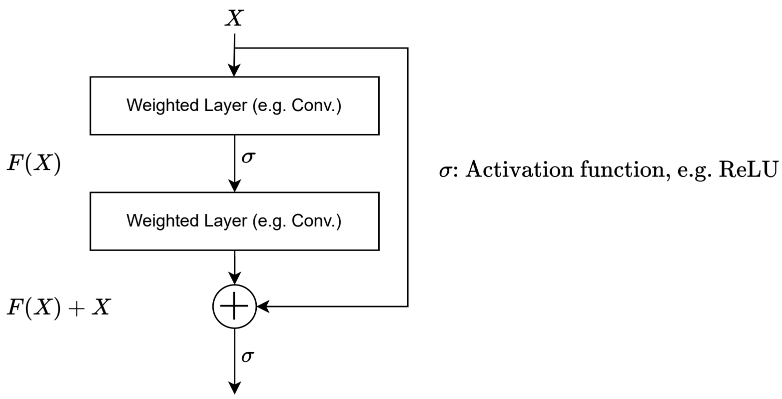


Figure 12: Example of a residual connection. This figure was adapted from He et al. [122].

Residual connections address the vanishing gradient problem by allowing gradients to flow unhindered to lower layers in the networks. Further, residual connections may simplify the process of learning an optimal mapping at each network layer, as it preconditions the layer to an identity mapping, which can easily be learned by setting the weights of F to 0 [122]. In practice, an optimal mapping may be closer to an identity mapping then a mapping given by a random initialization of F. Thus, the optimal mapping may be learned more easily by adding a perturbation to the identity mapping via $F(X) + X$. Residual connections have been shown to improve the state-of-the-art performance on various vision tasks and enable the training of models with hundreds of layers [122]. RBPNet [3] makes extensive use of residual connection, which envelope each dilated convolutional block.

5 INTERPRETING NEURAL NETWORKS

While the prediction of biomolecular properties is one of the primary goals of computational modeling in molecular biology research, the interest may extend beyond just producing accurate predictions. Understanding the process behind these predictions is equally crucial for gaining insights into biological processes. As a learned model generally represent an abstraction of some biological process, such as the recognition of an RBP-binding site on an RNA sequence, carefully studying the model and its predictions may give insights into those processes. In the following chapter, an overview over several model interrogation strategies is given.

5.1 ARE NEURAL NETWORKS BLACK BOXES?

A long-standing and prevailing view is that (deep) neural networks are so-called "black boxes" and are inherently difficult to interpret. Literature on interpretability in the context of machine learning is often accommodated with a depiction of a "complexity-interpretability" tradeoff [124], akin to the well-known bias-variance trade-off, where linear models and decision trees generally rank high on the interpretability axis and neural networks usually rank last, preceded by models such as Random Forests [125] and Support-Vector Machines (SVMs) [126] (Figure 13). Indeed, to improve the interpretability of neural networks, efforts have been made towards the development of hybrid models that first employ neural networks for feature extraction, which are then combined linearly [127, 128]. However, an alternative view is that while linear models and decision trees are usually applied to problems where the existence of linear or rule-based explanations are expected, neural networks are instead predominantly applied in situations where the relations between inputs and outputs are complex and difficult to formulate or visualize. Thus, the difficulty of interpreting neural networks may therefore be attributed to the complexity of the problem itself, rather than the nature of the model.

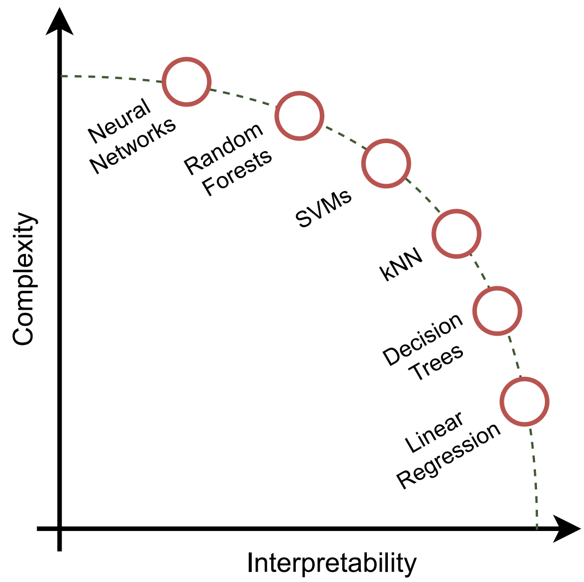


Figure 13: Complexity-interpretability tradeoff in machine learning. This figures was adapted from Arrieta et al. [124].

Considerable efforts have been made in recent years towards improving the interpretability of neural networks. While the development of interpretability methods has largely been spearheaded in the domains of computer vision and image recognition [129, 130], model interpretation is now commonly applied to study the mechanisms of (post-)transcriptional regulations, such as open chromatin [4], TF-binding [112, 114] and RBP-binding [1, 3] prediction. In the following, a brief overview of explainability methods for neural networks is provided, with references made to their application in the context of nucleotide-sequence prediction tasks, in particular protein-RNA interaction prediction.

5.2 LOCAL VERSUS GLOBAL EXPLAINABILITY

Explainability may be performed on a *local* or *global* level. While local feature importance aims to explain the decision-making processes of the model for a single data point, global importance assesses the importance of a feature with respect to the entire dataset. In the realm of RBP-binding prediction, local importance methods assign scores to individual nucleotides or sub-sequences within a single RNA sequence. Instead, global importance techniques may leverage these local scores to identify globally important sub-sequences found in the majority of RBP-binding RNA sequence, thereby constructing RBP-binding motifs that elucidate protein binding specificities across the dataset.

5.3 MODEL-FREE VERSUS MODEL-DEPENDENT EXPLAINABILITY

Model interpretation methods may be classified into model-free and model-dependent approaches. Model-free methods make no assumption on the underlying model and the model is instead considered to be a black box. Model-free methods operate by sending perturbed inputs through the model and observing their effect on the model output, thereby inferring feature importance.

In contrast, model-dependent methods are exclusively applicable to a certain class of methods, such as those trained via gradient-descent. Other examples include the direct interpretation of convolutional kernels in context of CNNs (Section 5.5.1) or the interpretation of attention maps for architectures that utilize attention mechanisms (Section 5.5.2). Besides neural networks, another instance of a model-dependent interpretation method is the *Gini-importance* in the context of Random Forests [125], which represents the average gain in class purity for decision-splits on a given feature.

5.4 FEATURE PERTURBATION AND OCCLUSION

Perturbation-based methods are a popular choice in cases where the model function is too complex for a direct investigation model parameters, such as model ensembles or neural networks. Assuming that the perturbation of important features will have a greater effect on the model's output than the perturbation of unimportant ones, perturbation-based methods systematically alter input features and record the effect on a model's output [131]. Feature importance may then be assigned by quantifying the change of the perturbed model output with respect to the unperturbed reference, for instance by computing the relative increase of the model's error. While perturbation-based methods generally make no assumption on the underlying model and access only its input and output, perturbation strategies may vary depending on the problem at hand. For instance, in tabular datasets a feature column may be shuffled, thereby destroying the feature information but preserving the feature distribution. Subsequently, the algorithm's drop in performance may be evaluated across the entire dataset to obtain a global (i.e. dataset-wide) feature importance score.

For image classification tasks, parts of an image, for instance individual or patches of pixels, may be *occluded* by setting their value to zero across all channels, yielding a local feature importance or "attribution" map for a given image [129] at varying resolution. Occlusion may be applied to a variety of tasks, albeit with task-specific occlusion values. For instance, in the context of DNA or RNA sequence classification, the input sequence is commonly occluded by replacing one-hot encoded bases with 0-vectors. In the context of protein-RNA interaction prediction, the impact of hypothetical SNVs on RBP-binding may be assessed in silico by perturbing individual positions in the input RNA sequence [1, 3].

Perturbation-based methods exhibit several limitations. First, the systematic perturbation of features can incur significant computational cost, especially when generating high-resolution local attribution maps. As the number of perturbations required is proportional to the number of input features for each sample, perturbing and scoring inputs with large models can become prohibitive. Second, perturbation may yield inconclusive results on feature importance in certain situations. This is exemplified in Figure 14, where $y = f(x_1, x_2) = 1 - max(0, 1 - x_1 - x_2)$ and $x_1, x_2 \in \{0, 1\}$. For $x_1 = x_2 = 1$, perturbing either x_1 or x_2 independently does not incur changes in the output of f, which may lead to the false conclusion that neither x_1 or x_2 are predictive for y.

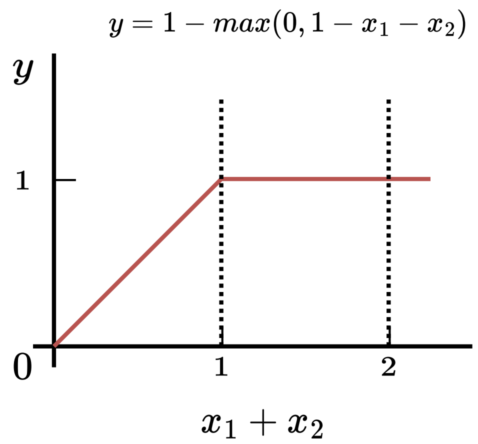


Figure 14: Limitations of perturbation-based explainability methods. For $x_1 = x_2 = 1$, the output y does not change as a result of perturbing x_1 or x_2. Consequently, the perturbation-importance of both input variables is 0. This figure was adapted from Shrikumar et al. [132].

5.5 ARCHITECTURE-SPECIFIC METHODS

Architecture-specific approaches are solely applicable to specific neural network architectures and thus represent an instance of model-dependent interpretability methods. Here, two architecture-specific approaches are presented — direct interpretation of convolutional kernels and attention maps.

5.5.1 *Interpreting Convolution Kernels*

CNNs decompose the identification of complex input features into the detection of multiple sub-features, enabling them to solve challenging tasks. The same paradigm may be applied for the interpretation of CNNs. That is, to understand why a CNN performs a particular prediction for a given input, one may investigate which low-level features are learned by its convolutional filters. Directly investigating the learned features of convolution kernels was first explored in the context of image classification, where multiple studies showed that early layers learn simple feature such as horizontal and vertical edges, while deeper layers learn more complex concepts, such as facial features [129, 130].

When applying CNNs to tasks with RNA or DNA sequence inputs, an important observation is that a motif PWM (Section 3.3) of length L can be represented by a single 1-dimensional convolutional kernel of the same length. Consequently, a single-layer CNN may act as an extractor for a set of task-specific sequence motifs, such as RBP-binding motifs. Several methods analyze first-layer kernels to gain insight into sequence motifs that govern the network's predictions [104, 114, 115, 133–135]. A common approach, popularized by DeepBind [114], builds an initial PFM by extracting (and subsequently aligning) sub-sequences from training or test-set samples which incur a high kernel activation. More specifically, given the i'th kernel K_i of length L and a sequence S, the method first identifies the position in the input sequence that yields the highest kernel activation $j = argmax_k K_i(S_k)$ and subsequently extracts the corresponding sub-sequence $S_{j-L+1,...,j}$. Repeating this process across many sequences results in a set of high-activation sub-sequences, which are then aligned to construct a kernel's PFM. Prior to PFM construction, DeepBind filters sub-sequences with an activation lower than 0, in order to remove uninformative sequences which show no particular association with the target kernel. A similar kernel interpretation approach is utilized by Trabelsi et al. [134] in the context of DeepRAM, which only extract sub-sequences from test-set sequences. Basset [115] applies a stricter sequence activation cutoff such that only sequences are used for PFM construction which activate the kernel by at least half of its maximum possible activation, given that sequences are one-hot encoded. iDeepS [133] extends this approach to the interpretation of RNA secondary structure binding motifs, using a one-hot encoded secondary structure alphabet. Similar to Basset, Pysster [104] re-defines the activation cutoff by first computing the average maximum kernel activation across all sequences for each class. The highest average maximum kernel activation across all classes then serves as the activation cutoff for incorporating sub-sequences in the PFM of the target kernel. Notably, Budach [136] further extends the analysis of first-layer kernels by investigating co-occurring kernels via clustering, in order to identify proteins that exhibit different modes of binding or bind the DNA or RNA sequences via different binding domains [136]. However, they observe that correlated kernels often learn to recognize highly similar motifs, which may be due to forced redundancy induced by Dropout regularization.

A limitation of deriving binding motifs from first-layer CNN kernels is that it is based on the assumption that proper motifs are learned in the first layer. As deep neural networks detect features hierarchically, kernels may only learn partial motifs which are then aggregated in deeper layers. Thus, investigation of first-layer kernels may not capture the overall binding footprint of the target protein. Further, investigating first-layer kernels ignores feature dependencies. For instance, co-occurrence of two or more motifs at a specific spacing may determine the classification of a given sequence and the investigation of kernels independently may fail to uncover such motif cooperativity. While this is partially addressed by Budach [136] via investigation of joint kernel activation, the search space grows exponentially in the number of considered kernels, rendering this approach intractable for investigating the cooperativity of several kernels.

5.5.2 *Attention Maps*

The attention mechanism is an integral part of several state-of-the-art models for natural language and computer vision tasks [137–141]. Given a set of token representations (also referred to as *embeddings*), the representations are first linearly projected to a set of query (Q), key (K), value (V) vectors. Next, the compatibility between a each query and key vector is evaluated, typically using a scaled dot-product similarity, resulting in attention scores (Eq. 10).

$$\text{Attention Scores} = \frac{QK^T}{\sqrt{d_k}}, \text{ where } d_k \text{ is the dimensionality of the key vectors.} \tag{10}$$

The scores are subsequently normalized into attention weights using a softmax function (Eq. 11), signifying the relevance of a particular key to a specific query. Given the attention weights, the weighted sum of the associated value vectors is computed, resulting in an attentive representation.

$$\text{Attention Weights} = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right) \tag{11}$$

In Transformer models [142], multi-head attention expands this process by projecting Q, K, and V through different linear transformations, allowing the model to discern and incorporate different patterns in the input. For further details, such as the use of positional encodings, the reader may refer to [142].

Since attention weights represent the relative contribution of input tokens towards an output token, they may be interpreted as importance scores. Commonly, transformer-based models make use of a special class-token (CLS), which is prepended to the input and whose final-layer embedding vector may serve as input to a task-specific output head during supervised fine-tuning. To identify important input tokens for the fine-tuned task, attention weights of the CLS token can be computed, which yields a feature importance vector across all tokens. Attention-based importance analysis is utilized by DNABERT [143] and BERT-RBP [144] to identify important DNA or RNA sub-sequences for the prediction of TF-or RBP-binding, respectively.

5.6 gradient-based scoring methods

Gradient-based methods encompass a large class of feature importance methods that operate by computing the gradients of the input features with respect to the network function f [129, 130, 145]. In context of classification tasks, gradient-based approaches assign an importance score to each feature x based on the change in the predicted probability score of a target class label y, as a result of an infinitesimal change in x. While not strictly architecture-specific, gradient-based approaches require that the underlying model is differentiable and are therefore popular for the interpretation of neural networks.

5.6.1 *Saliency Maps and Gradients * Input*

Saliency maps (also known as "Vanilla Gradients" [146]) were first proposed by Simonyan et al. [130] in the context of image classification. Given an input x and a neural network f which predicts some scalar y, for instance a class-specific probability score, such that

$$y = f(x) \tag{12}$$

the prediction y can be linear approximated around x via the first-order Taylor expansion

$$f(x) \ = w^T x + b \tag{13}$$

where w is the derivative of f [130]. Simonyan et al. then define the attribution of a feature x_i as $|w_i|$, that is the absolute of the gradient of feature x_i with respect to y. For visualization purposes, a single attribution value is assigned to each pixel in the input by taking the maximum value across all RBG channels. In the context of one-hot encoded RNA or DNA sequence, the single position-wise attribution value is naturally given by the value of the set nucleotide, i.e. the nucleotide present at the given position. Saliency maps as proposed by Simonyan et al. [130] may yield noisy feature attribution maps, as neural network gradients are often subject to sharp local fluctuations in the input [147]. Further, while the absolute gradient may quantify the importance of a feature, the strictly positive attribution value does not give insight into the positive or negative contribution of the feature towards the prediction [131]. To partially address these shortcomings, Shrikumar et al. [148] propose to multiply the gradients of the input with the input itself. This preserves the sign and generally sharpens the attribution maps, as high-gradient features with a high feature values are up-weighted. For one-hot encoded inputs, this produces attribution maps which correspond to a masked version of the input gradients.

Both Saliency maps and Gradients * Input require a single forward-and backward-pass through the network and are thus less costly to compute than importance scores generated via feature perturbation or occlusion (Section 5.4). They are employed for interpreting various models that predict properties of nucleotide sequences, including ChromTransfer [4] and Enformer [149]. Furthermore, RNAProt [150] utilizes Saliency maps for constructing RBP-binding motifs.

5.6.2 *SmoothGrad*

The highly non-linear nature of neural networks leads to complex and rugged gradient landscapes, which may lead to networks exhibiting *noisy* gradients — abrupt changes in gradient values due to small alterations in input features [147]. Based on the assumption that the gradient of f at a given point in the input is less informative than a local average of gradient values, Smilkov et al. [147] propose to compute attribution scores by smoothing gradients around the input. Formally, instead of computing gradients of an input x directly via $M(x) = df(x)/dx$, a stochastic approximation of the local average is computed via

$$\tilde{M}(x) = \frac{1}{n}\sum_{1}^{n} M(x + \mathcal{N}(0, \sigma^2)) \tag{14}$$

That is, a given input is augmented n times by addition of Gaussian noise with standard deviation σ. The gradients of each noise-perturbed input are then computed and averaged, to produce the final SmoothGrad attribution scores. While this enhances the robustness of the feature importance estimate, it also results in an n-fold increase in computational cost compared to Saliency maps. SmoothGrad is employed in the context of PrismNet [151] to identify nucleotide positions contributing to RBP-binding prediction for a target protein.

5.6.3 *Integrated Gradients*

While SmoothGrad stabilizes gradients with respect to local fluctuations, it is still subject to saturation effects, which are a common issue in gradient-based attribution methods. Informally, gradient saturation occurs when an input feature assumes a value that is beyond a threshold necessary to produce a certain model output, after which an increase in the feature's value does not change the model output in a significant way. This is demonstrated in Figure 15.

For $x \in [0, t]$, an increase in the feature value incurs a linear increase in the model output and thus a gradient of 1. However, after some threshold t the feature becomes saturated, such that further increasing x beyond t does not increase the model's output. Consequently, the gradient of feature x is 0 for $x > t$. Computing feature attribution via Saliency maps or SmoothGrad in a narrow neighborhood of x would thus wrongly suggest that x is non-informative. To address this issue, Sundararajan et al. [145] developed Integrated Gradients (IG), which, rather than computing gradients at or around a given input, averages gradients along a path from a specified baseline to the input. Formally, let $f : \mathbb{R}^n \to [0, 1]$ be a neural network with a scalar output and $x \in \mathbb{R}^n$ a target input for which we would like to compute attributions. Further, let x' be a baseline input that is chosen manually, such as a null-tensor in the context of one-hot encoded RNA or DNA sequences. Then, the IG attribution map for input x is defined as

$$IG(x) := (x - x') \times \int_{\alpha=0}^{1} \frac{\partial f(x' + \alpha \times (x - x'))}{\partial x} d\alpha \tag{15}$$

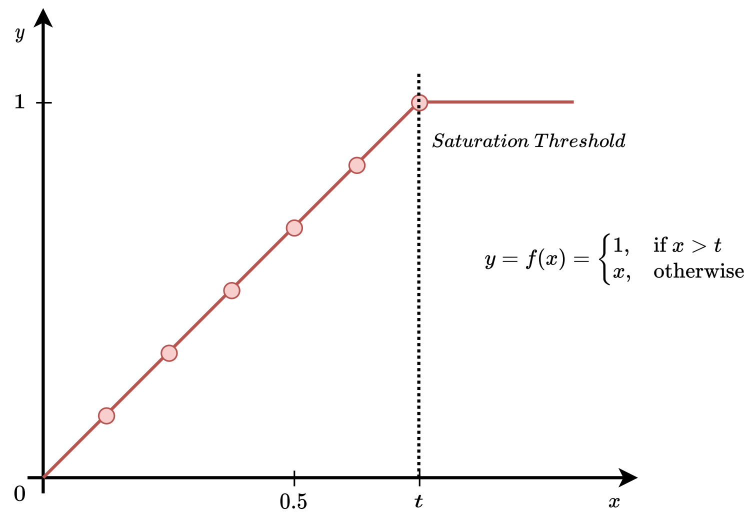


Figure 15: Up to a threshold t, increasing x results in a linear increase in y. However, increasing x beyond a threshold t results in saturation, with no further increase in y.

That is, IG attributions are defined as the integral of gradients along the linear path from x' to x. As IG accumulates gradients along the entire path from the baseline to the target input, IG overcomes potential saturation effects [145] present in Saliency maps or SmoothGrad. Further, IG satisfies the *completeness* property, that is, attribution scores sum up to the difference between the output of f for the baseline x' and the target input x. Similar to SmoothGrad, IG is associated with an increased computational burden compared to Saliency maps, as estimating the integral in 15 via Riemann approximation requires one to sample a sufficient number of inputs along the linear path from baseline to target input. Sundararajan et al. [145] note that in practice, between 20 to 300 steps are required for a reasonable approximation. Additionally, the choice of the baseline input significantly impacts the resulting attribution scores and should be selected carefully. Commonly, baselines are designed such that they represent the absence of any task-specific signal, such as a black or white-noise image. However, designing baselines may be difficult for some input modalities, such as icSHAPE [152] scores, which are used in the context of PrismNet [151].

Several protein-RNA interaction prediction methods make use of IG for identifying RBP-binding motifs, including DeepRiPe [153], Multi-resBind [154] and MultiRBP [155]. In the core publication [1], IG is used to identify binding motifs of human RBPs to the SARS-CoV-2 RNA sequence. Notably, the core publication presenting RBPNet [3] uses a modified version of IG which enables attribution of vector-valued outputs.

5.7 DEEPLIFT

DeepLIFT [132] aims to explain the output of a target neuron y, computed from inputs $x_1, ..., x_n$, as the difference from some reference input $x_1^0, ..., x_n^0$ with reference activation y^0. In other words, the difference in the target neuron's activation, $\Delta y = y - y^0$, is explained in terms of the difference between the target and reference inputs by assigning contribution scores $C_{\Delta x_i \Delta y}$ to each x_i such that

$$\sum_{i=0}^{N} C_{\Delta x_i \Delta y} = \Delta y \tag{16}$$

Scores may further be separated into positive and negative contributions via $C_{\Delta x \Delta y} = C_{\Delta x^+ \Delta y} + C_{\Delta x^- \Delta y}$. To compute contribution scores, Shrikumar et al. [132] define a *multiplier* $m_{\Delta x \Delta y}$ as

$$m_{\Delta x \Delta y} = \frac{C_{\Delta x \Delta y}}{\Delta x}, \tag{17}$$

which represents the effect of Δx on Δy, scaled by Δx. To propagate contribution scores from a target unit to the network input across several (hidden) layers, Shrikumar et al. defined the *chain rule for multipliers* as

$$m_{\Delta x_i \Delta y} = \sum_j m_{\Delta x_i \Delta h_i} \times m_{\Delta h_i \Delta y} \tag{18}$$

with $h_i, ..., h_n$ being the units of a hidden layer h. This allows computation of multipliers for all network units with respect to the target unit in an efficient manner via backpropagation. For non-linear transformations, Shrikumar et al. introduce the *Rescale rule*. That is, given a non-linear unary function $y = f(x)$, it follows that $C_{\Delta x \Delta y} = \Delta y$ and $m_{\Delta x \Delta y} = \frac{\Delta y}{\Delta x}$ (Eq. 16). This overcomes the thresholding problems associated with *Saliency Maps* and Gradients * Input attribution methods (Section 5.6.1), as depicted in Figure 16.

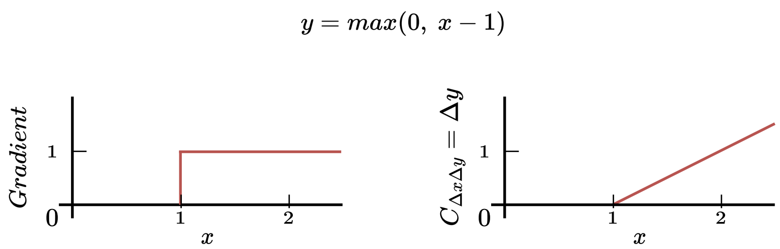


Figure 16: Gradient discontinuity problem for the ReLU activation function for input x and bias -1. While gradients show a discontinuity when moving from $x = 1$ to $x = 1 + \epsilon$, DeepLIFT scores increase continuously such that at $x = 1 + \epsilon$, $C_{\Delta x \Delta y} = \epsilon$ [132].

As for Integrated Gradients, the choice of the reference input is crucial for the performance of DeepLIFT. In the context of DNA or RNA modeling tasks, Shrikumar et al. [132] propose reference inputs containing expected nucleotide frequencies or shuffled version of the input sequence as viable options. DeepLIFT was applied by Avsec et al. [112] to identify binding motifs of transcription factors with BPNet. Note that in its current implementation[1], DeepLIFT is exclusive to Tensorflow 1. Therefore, it is currently not applicable to neural network implementations on the basis of Tensorflow 2 or PyTorch.

1 https://github.com/kundajelab/deeplift (Accessed on September 19, 2023)

6 APPLICATIONS OF DEEP LEARNING MODELS IN GENOMICS

Due to its success in domains like computer vision and natural language processing, deep learning has become integral to problem-solving in computational biology, including genomics and transcriptomics. For the analysis of raw sequencing data, Oxford Nanopore employs deep learning methods to improve base calling accuracy for long-read sequencing data [156], while Deep Variant [157] performs variant calling based on read pileups around candidate SNVs. To analyze functional aspects of the genome, methods have been developed to predict binding of transcription factors to DNA [158], chromatin state [4, 115], epigenetic histone marks [159] and DNA methylation [160]. To gain insights into RNA chemistry, deep learning methods have been proposed to improve computational folding of RNA into 2D [161] and 3D [162] structures over current energy-based algorithms. Machine-and deep-learning methods have been utilized to study various aspects of RNA processing, such as the prediction of splicing from RNA sequence [163]. Additional applications encompass the prediction of RNA half-life from sequence [164], RNA localization prediction [165], and translation efficiency prediction [166]. Finally, the prediction of RBP-binding represents an essential application of deep learning in computational RNA biology, with dozens of methods proposed in recent years [2]. Many of the aforementioned methods are classification-based and predict discretized functional properties. For instance, Basset [115] predicts whether the chromatin for a given sequence window is open or closed, DeepHistone [159] predicts whether a given histone mark is present or absent and DeepRiPe [153] predicts whether a given RNA sequence is bound by an RBP. To overcome this simplistic, dichotomic view, recent methods aim to directly model the signal generated by genomic assays. For instance, BPNet [112] directly models read counts of ChIP-nexus experiments, which is extended to ATAC-seq and eCLIP data for the study of chromatin state and RBP-binding by ChromBPNet [113] and RBPNet [3], respectively. Recently, models such as Basenji [167] or Enformer [149] model data across various genomic assays at a massively multi-task scale and across distances of up to 200kb, albeit at lower nucleotide resolution. Another recent trend is the development of large transformer-based models, which are trained in a self-supervised manner, often via masked-language modeling. Examples of this family of models include DNABERT [143], Nucleotide Transformer [168], HyenaDNA [169] and others [118, 170, 171]. Rather than directly solving a genomics tasks, these models aim to learn meaningful internal representation of DNA and RNA sequences, to serve as a foundation for future downstream tasks. While current iterations of nucleotide language models have not achieved the same level of success and widespread adoption as protein language models like ESM [172] and ProtTrans [173], they are anticipated to gain significant attention in the upcoming years.

Perhaps one of the most crucial applications of machine learning and deep learning methods in genomics is their use in quantifying the impact of genetic variants, commonly referred to as *variant impact scoring*. Variant impact scoring plays a pivotal role in the functional annotation of disease-associated variants, which are commonly obtained through genome-wide association studies. In many cases, machine learning models allow for a direct variant impact scoring by comparing the prediction of the perturbed reference and unperturbed DNA or RNA sequence. For instance, methods such as DeepSEA [174] have been developed, which predict the effects of non-coding variants by quantifying their effect on the chromatin profile, including TF binding, histone mark and chromatin accessibility. On the other hand, recent transformer-based protein language models have been utilized to predict the impact of coding variants [175]. On the transcript-level, tools like AbSplice [30] or CADD-Splice [176] predict variants associated with aberrant splicing. Park et al. utilized Seqweaver [177], a model trained to predict protein-RNA interactions, to unveil the role of deregulated RBP target sites as a significant risk factor for psychiatric disorders. Saluki [164] predicts mRNA half-life from sequence and was used by Agarwal et al. to estimate the effect of sequence variants on mRNA stability. Kelley [167] employed Basenji to classify Mendelian disease variants from HGMD [178] and ClinVar [179], as well as fine-mapped variant-disease associations from GWAS. Enformer [149] has been successful in discriminating likely causal from spurious eQTL variants. Recently, Linder et al. [180] developed Borzoi, an Enformer-esque model, which predicts RNA-seq coverage across multiple cell lines and tissues. Apart from classifying eQTL-associated variants successfully, the authors demonstrated that Borzoi can isolate variant impacts across multiple layers of regulation, including splicing and polyadenylation. Finally, large nucleotide language models have been shown to enable zero-shot variant impact scoring [118]. Collectively, these models provide valuable insights into the functional impact of sequence variants across diverse cellular processes, the identification of causal relationships between genetic variants and disease phenotypes, and the discovery of novel drug targets.

Part IV

ACADEMIC CONTRIBUTIONS

AIMS AND SCOPE OF ACADEMIC CONTRIBUTIONS

RBPs are known host factors in SARS-CoV-2 and other viral infections (Section 2.2), however, resolving precise binding locations of human RBPs on the viral genome experimentally is costly and time consuming, due to CLIP-seq protocols being limited to investigating one protein at a time (Section 3.2). Yet, knowledge of binding sites is crucial to address question such as: *Which RBPs bind the viral genome and where? Are RBP binding sites conserved in the viral sequence? Have newly emerging viral strains acquired gain-or loss-of-binding mutations that may affect viral fitness?* Core publication [1] addresses these questions computationally, by leveraging deep learning models trained on human CLIP data to predict protein-RNA interactions.

An abundance of deep learning methods for protein-RNA interaction prediction have been developed, however, method evaluation is usually performed on heterogeneous datasets that vary between studies in terms of CLIP protocol, RBPs and data processing steps. Consequently, it is difficult to answer questions such as: *Which methods are the state-of-the-art? Which model design choice, such as model architecture, positively affect performance? Which additional input modalities, such as RNA secondary structure, improve model performance? Does model performance vary across CLIP protocols?* Publication [2] addresses these questions by establishing a benchmark and evaluating several approaches across a large number of CLIP datasets.

Existing methods for protein-RNA interaction prediction, including those reviewed in core publication [2], predict if — not where — an input sequence is bound by an RBP of interest. As practical application often require predictions with high spatial resolution, researchers need to employ computationally costly scanning approaches, as done in core publication [1]. CLIP protocols such as iCLIP and eCLIP measure protein-RNA interaction at single-nucleotide resolution (Section 3.2.1), which is not leveraged by current methods. Thus, a natural question to ask is: *How can existing computational approaches be modified and extended to leverage the resolution provided by single-nucleotide CLIP protocols?* To this end, core publication [3] proposes RBPNet, a deep learning method which directly models the raw CLIP signal at nucleotide resolution.

7 BINDING PREDICTION OF HUMAN RBPS TO SARS-COV-2

A computational map of the human-SARS-CoV-2 protein–RNA interactome predicted at single-nucleotide resolution

Authors: Marc Horlacher, Svitlana Oleshko, Yue Hu, Mahsa Ghanbari, Giulia Cantini, Patrick Schinke, Ernesto Elorduy Vergara, Florian Bittner, Nikola S Mueller, Uwe Ohler, Lambert Moyon, Annalisa Marsico

Journal: NAR Genomics and Bioinformatics, 2023 [1]

Summary: The COVID-19 pandemic claimed the lives of millions while profoundly affecting the lives of billions. Despite successful vaccination campaigns, treatment of COVID-19 post-infections remains challenging, particularly for individuals with compromised immune systems, highlighting the need for novel therapeutic avenues. While RNA-binding proteins are well-known host factors in the context of viral infection, determining the permissiveness of a cell towards sustained viral replication, comprehensive mapping of human RBP-viral RNA interactions is challenging due to experimental constraints and viral sequence variations. This study addresses this knowledge gap by leveraging deep learning models (Pysster and DeepRiPe) trained on a wide array of human RBP-RNA interaction datasets. Once trained, these models are applied to predict binding sites of human RBPs on the SARS-CoV-2 RNA, thereby generating the a comprehensive map of predicted host RBP-viral RNA interactions at near single-nucleotide resolution. To ensure that predictions are of high quality, model performance is estimated on human transcripts and low-performing models are removed by applying stringent performance thresholds. Further, null-models are established via prediction on randomized viral sequences and predictions are validated by comparing them to previously reported interactions on SARS-CoV-2. By extending predictions to six human pathogenic coronaviruses, conserved and differential bound RBP binding sites in the untranslated regions of SARS-CoV-1, SARS-CoV-2, and MERS are identified. Using *in silico* mutagenesis, the study examines sequence mutations within 11 SARS-CoV-2 variants-of-concern and their predicted impact on RBP-RNA interactions, shedding light on potential alterations in viral fitness. Further, results are integrated with clinical data, including Genome Wide Association Studies (GWAS), CRISPR studies and patient OMICS data, highlighting RBPs with clinical relevance and offering valuable insights into potential therapeutic targets. In conclusion, this study establishes a comprehensive catalogue of predicted host-virus RBP-RNA interactions for SARS-CoV-2, providing a valuable resource for further investigations into the virus's biology and potential therapeutic strategies. The findings contribute to unraveling the intricate interplay between the virus and host cellular machinery, ultimately aiding in the fight against COVID-19 and future coronavirus pandemics.

Availability: Code to replicate the results of this study is available at `https://github.com/mhorlacher/sc2rbpmap`. To facilitate an easy access to the results of this study, a dashboard web-sever of predictions and variant impact scores is available at `https://sc2rbpmap.helmholtz-muenchen.de`.

Contributions: Study design and conceptualization; Implementation of training and evaluation scripts for the Pysster model; Gathering and preprocessing of eCLIP data from the ENCODE database; Training of Pysster model on the Helmholtz Center's HPC cluster; Implementation and performance analysis of the model's "performance in practice"; Scanning of viral sequences with Pysster models; Impact scoring of sequence variants from variants of concern and *in silico* perturbation of the entire SARS-CoV-2 sequence; Figure design, in particular figures 2, 3 and 5; Preparation of the manuscript text and formatting of the submission.

Published online 20 February 2023 *NAR Genomics and Bioinformatics, 2023, Vol. 5, No. 1* **1**
https://doi.org/10.1093/nargab/lqad010

A computational map of the human-SARS-CoV-2 protein–RNA interactome predicted at single-nucleotide resolution

Marc Horlacher[1], Svitlana Oleshko[1], Yue Hu[1,5], Mahsa Ghanbari[2,3], Giulia Cantini[1], Patrick Schinke[1], Ernesto Elorduy Vergara[1], Florian Bittner[4], Nikola S. Mueller[4], Uwe Ohler[2,3], Lambert Moyon[1,*] and Annalisa Marsico[1,*]

[1]Computational Health Center, Helmholtz Center Munich, Munich, Germany, [2]Institutes of Biology and Computer Science, Humboldt University, Berlin, Germany, [3]Max Delbruck Center, Computational Regulatory Genomics, Berlin, Germany, [4]Knowing01 GmbH, Munich, Germany and [5]Informatics 12 Chair of Bioinformatics, Technical University Munich, Garching, Germany

Received June 27, 2022; Revised January 10, 2023; Editorial Decision January 13, 2023; Accepted February 14, 2023

ABSTRACT

RNA-binding proteins (RBPs) are critical host factors for viral infection, however, large scale experimental investigation of the binding landscape of human RBPs to viral RNAs is costly and further complicated due to sequence variation between viral strains. To fill this gap, we investigated the role of RBPs in the context of SARS-CoV-2 by constructing the first *in silico* map of human RBP-viral RNA interactions at nucleotide-resolution using two deep learning methods (pysster and DeepRiPe) trained on data from CLIP-seq experiments on more than 100 human RBPs. We evaluated conservation of RBP binding between six other human pathogenic coronaviruses and identified sites of conserved and differential binding in the UTRs of SARS-CoV-1, SARS-CoV-2 and MERS. We scored the impact of mutations from 11 variants of concern on protein–RNA interaction, identifying a set of gain- and loss-of-binding events, as well as predicted the regulatory impact of putative future mutations. Lastly, we linked RBPs to functional, OMICs and COVID-19 patient data from other studies, and identified MBNL1, FTO and FXR2 RBPs as potential clinical biomarkers. Our results contribute towards a deeper understanding of how viruses hijack host cellular pathways and open new avenues for therapeutic intervention.

INTRODUCTION

SARS-CoV-2, causative agent of the recent COVID-19 pandemic, has and still is affecting the lives of billions of people worldwide. Despite the large-scale vaccination effort, the number of infections and deaths remains high, primarily among the non-vaccinated and otherwise vulnerable individuals. Difficulty to control SARS-CoV-2 infections is partly due to the continuous emergence of novel viral variants, against which the full efficacy of current vaccines is still debated, as well as the lack of effective medication. This calls for a better understanding of the biology of SARS-CoV-2 to design alternative therapeutic strategies. SARS-CoV-2 is a betacoronavirus with a positive-sense, single-stranded RNA of 30kb (1). Upon infection, the released RNA molecule depends on the host cell's protein synthesis machinery to express a set of viral proteins crucial for replication (2). The genomic RNA is translated to produce non-structural proteins (nsps) from two open reading frames (ORFs), ORF1a and ORF1b, and it also contains untranslated regions (UTRs) at the 5′ and 3′ ends of the genomic RNA (1). A recent study revealed the complexity of the SARS-CoV-2 transcriptome, due to numerous discontinuous transcription events (3). Negative sense RNA intermediates are generated to serve as the template for the synthesis of positive-sense genomic RNA (gRNA) and subgenomic RNAs (sgRNA) which encode conserved structural proteins (spike protein [S], envelop protein [E], membrane protein [M] and nucleocapsid protein [N]), and several accessory proteins (3a, 6, 7a, 7b, 8 and 10) (3). During its life cycle, SARS-CoV-2 extensively interacts with host factors in order to facilitate cell entry, transcription of viral RNA and translation of subgenomic mRNAs, virion maturation and evasion of the host's immune response (1,4,5).

*To whom correspondence should be addressed. Tel: +49 89318749193; Email: lambert.moyon@helmholtz-muenchen.de
Correspondence may also be addressed to Annalisa Marsico. Tel: +49 89318743073; Email: annalisa.marsico@helmholtz-muenchen.de

Mechanisms of virus-host interaction are multifaceted and include protein–protein interactions (PPIs), binding of viral proteins to the host transcriptome (6), RNA–RNA interactions and binding of host proteins to viral RNAs. Studies on SARS-CoV-2 infected cells to date have predominantly focused on the entry of SARS-CoV-2 into human epithelial cells, which involves the interaction of the viral spike protein S with the human ACE2 receptor (3). Other studies characterized changes in the host cell transcriptome and proteome upon infection and identified host factors essential for viral replication via CRISPR screenings (7–9).

Lastly, mapping of protein–protein interactions (PPIs) between viral and host proteins has revealed cellular pathways important for SARS-CoV-2 infection. For instance, a recent study identified close to 300 host-virus interactions in the context of SARS-CoV-2 (8).

However, these studies have been of limited impact with respect to revealing how the viral RNA is regulated during infection.

RNA viruses hijack key cellular host pathways by interfering with the activity of master regulatory proteins, including RNA binding proteins (RBPs) (10). RBPs are a family of proteins that bind to RNA molecules and control several aspects of cellular RNA metabolism, including splicing, stability, export and translation initiation. Often, RNA targets of an RBP share at least one common local sequence or structural feature which facilitates the recognition of the RNA by the protein. Host cell RBPs have previously been reported to interact with viral RNA elements and influence several steps of the viral life cycle, such as recruitment of viral RNA to the membrane and synthesis of subgenomic viral RNAs (11,12,13). Indeed, in a recent proteome-wide study, 342 RBPs were identified to be annotated with gene ontology (GO) terms related to viruses, infection or immunity with a further 130 RBPs being linked to viruses in literature (13). Examples include the Dengue virus (14), the Murine Norovirus (MNV) (15) and Sindbis virus (SINV), where it has been shown that RBPs stimulated by the infection redistribute to viral replication factories and modulate the success of infection (13). The ability of viral RNAs to recruit essential host RBPs could explain permissiveness of certain cell types as well as its range of hosts (11), which is especially relevant for zoonotic viruses such as SARS-CoV-2. In the context of SARS-CoV infection, DEAD-box helicase 1 (DDX1) RBP has been shown to facilitate template read-through and thus replication of genomic viral RNA, while heterogeneous nuclear ribonucleoprotein A1 (hnRNPA1) might regulate viral RNA synthesis (5,16,17). Multiple recent studies show that SARS-CoV-2 RNAs extensively interact with both pro- and anti-viral host RBPs during its life cycle (18–21). Using comprehensive identification of RNA-binding proteins by mass spectrometry (ChIRP-MS), Flynn *et al.* (18) identified a total of 229 vRNA-bound host factors in human Huh7.5 cells with prominent roles in protecting the host from virus-induced cell death. Schmidt *et al.* (19) identified 104 vRNA-bound human proteins in the same cell line via RNA antisense purification and quantitative mass spectrometry (RAP-MS), with GO-terms strongly enriched in translation initiation, nonsense-mediated decay and viral transcription. The authors further confirmed the specific location of vRNA binding sites for cellular nucleic acid-binding protein (CNBP) and La-related protein 1 (LARP1) via enhanced cross-linking immunoprecipitation followed by sequencing (eCLIP-seq), which were both associated to restriction of SARS-CoV-2 replication (19). Lee at al. (20) identified 109 vRNA-bound proteins via a modified version of the RAP-MS protocol and linked those RBPs to RNA stability control, mRNA function, and viral process. Further, the authors showed 107 of those host factors are found to interact with vRNA of the seasonal betacoronavirus HCoV-OC43, suggesting that the vRNA interactome is highly conserved. Finally, Labeau et al. (21) used ChIRP-MS to identify 142 host proteins that bind to the SARS-CoV-2 RNA and showed, in contrast to Flynn et al. (18), that siRNA knockdown of most RBPs cellular expression leads to a significant reduction in viral particles, suggesting that the majority of RBPs represent pro-viral factors. Taken together, there is strong evidence that SARS-CoV-2, like other RNA viruses, heavily relies on the presence of a large number of essential RNA-binding host factors. However, the sets of SARS-CoV-2 relevant RBPs from different studies have limited overlap and the outcome depends on the specific cell line utilized in the experiment. While the above methods allow for the identification of RBPs that interact with SARS-CoV-2 in the context of infection, they do not give insight into where these interactions occur along the viral RNA. Indeed, a comprehensive large scale analysis of the propensities of different host RBPs to bind to RNA elements across the SARS-CoV-2 genome is currently missing. This is a severe limitation, as prevalence and high spatial resolution of protein–RNA interaction sites is necessary for understanding the hijacking of the cellular post-transcriptional machinery by the virus.

Cross-linking and immunoprecipitation (IP) followed by sequencing (CLIP-seq) assays (22), including PAR-CLIP and eCLIP protocols, are the most widely used methods to measure RBP-RNA interactions *in vivo* at high nucleotide resolution and are able to provide sets of functional elements that are directly bound by an RBP of interest (23). While CLIP-seq experiments allow for precise identification of host factor interaction with viral RNAs, the high cost of profiling interactions across a large number of RBPs becomes prohibitive at larger scales, as dedicated pull-down and sequencing has to be performed for each RBP individually. Therefore, such datasets have been generated only for a small number of proteins on SARS-CoV-2 (19). Further, in order to keep up with the continuous emergence of novel SARS-CoV-2 variants, CLIP-seq experiments would need to be repeated for the genome of each viral strain in order to account for (or to identify) gain- or loss-of-binding variants. Recent advances in machine and deep learning have enabled a cheaper but powerful alternative by computationally modeling the binding preference of RBPs using information from existing CLIP-seq datasets, such as those generated as part of the ENCODE project (24).

In this study, we present a step towards filling the gap of missing spatial information of human-SARS-CoV-2 protein–RNA interactions by predicting interaction sites computationally at nucleotide resolution. We train and optimize two recent Convolutional Neural Network (CNN) based methods, Pysster (25) and DeepRiPe (26), on hundreds of human eCLIP and PAR-CLIP datasets and use trained models to predict RBP binding on viral sequences. By that we provide, to our knowledge, the first comprehensive single-nucleotide resolution *in silico* map of host RBP-viral RNA interaction for SARS-CoV-2 as well as six other human coronaviruses and identify sequence mutations, which significantly alter RBP-RNA interaction across 11 different SARS-CoV-2 variants-of-concern. We recapitulate human RBPs, which are predicted or experimentally determined to binding to SARS-CoV-2 by previous studies and predict binding for host RBPs with no previously reported binding to SARS-CoV-2. We integrate knowledge of these proteins across other pathogens and highlight RBPs with clinical relevance, by annotating those that were found among SARS-CoV-2-associated genes from Genome Wide Association Studies (GWAS) (27), CRISPR studies (28–32), physical binding experiments (18,19,29), or patient OMICS data from blood serum and plasma (30–37). Finally, we perform extensive *in silico* single-nucleotide perturbations across the SARS-CoV-2 genome to identify mutations that would lead to gain-and/or loss-of predicted RBP binding sites and thus may alter viral fitness.

MATERIALS AND METHODS

The overall workflow of our approach is summarized in Figure 1, from model training, to the *in silico* mapping of the SARS-CoV-2 RBP-RNA interactome and downstream analysis. We first obtained binding site information of publicly available eCLIP experiments of 150 RBPs from the ENCODE (24) database and pre-processed them to obtain a set of high-quality sites of protein–RNA interaction. For each RBP, a convolutional neural network (CNN) classifier to predict the likelihood of RBP-binding to an arbitrary input RNA sequence was trained using the *pysster* (25) framework, resulting in 150 pysster models (Figure 1A). For RBPs not contained in the ENCODE dataset, we included DeepRiPe (26) models pre-trained on 59 PAR-CLIP datasets. Next, we performed extensive model performance evaluation on custom trained pysster models and removed poorly performing models from downstream analysis.

Using high-quality models, we predicted the probability of each RBP binding to individual nucleotides in the SARS-CoV-2 genome using a sliding-window scanning approach (Figure 1B). In addition, we controlled for false positive hits by computing, for each RBP and each single-nucleotide binding prediction, an empirical p-value using a randomized background sequence and retained only significant hits. Subsequently, consecutive high-scoring and significant positions were aggregated into larger binding-site regions. We thus constructed a comprehensive *in silico* binding map of human RBPs on the SARS-CoV-2 genome and clustered predicted RBP binding sites across different viral genomic regions to unravel potential regulatory patterns (Figure 1B). Exploiting the capability of CNNs to learn complex sequence patterns, we identified known binding motifs at predicted RBP binding sites. Finally, we utilized our models to score the impact of sequence variants identified in 11 viral variants of concern (Figure 1C) and identified conserved and novel predicted binding sites across 6 other coronaviruses, including SARS-CoV-1 and MERS (Figure 1D).

ENCODE data and preprocessing

Enhanced CLIP (eCLIP) datasets were obtained from the ENCODE project database, which comprises 223 eCLIP experiments of 150 RBPs across two cell lines, HepG2 and K562. For RBPs with experiments in both cell lines, we selected only data of eCLIP experiments from the HepG2 cell line for downstream analysis. Narrow peaks of each eCLIP library were taken directly from ENCODE and preprocessing was performed as follows: for each of the two replicates of a given eCLIP experiment, peaks were first intersected with mRNA locations obtained from the GENCODE database (Release 35) and only overlapping peaks were retained. Next, the 5′-end of each peak was defined as the cross-linked site, as it usually corresponds to the highest accumulation of reverse transcription truncation events. A 400nt window was then centered around the cross-linked site for each peak, defining the input window of the downstream model. Input windows of both replicates were intersected reciprocally with a required overlap fraction of 0.75, ensuring that only those peaks which are present in both replicates are considered for downstream training set construction. Finally, the top most 50 000 windows with a read-start count FC of 2.0 above the control (SMInput) experiment were selected for each RBP.

Pysster training set construction

For each RBP, a classification dataset of bound (positive) and unbound (negative) RNA sequences was constructed. Positive samples were obtained by taking corresponding 400nt peak-region windows from the previous step, while two distinct sets of negative samples were generated. First, 400nt long regions which did not overlap with CLIP peaks of the given RBP were sampled from transcripts harboring at least one CLIP peak. This constraint ensures that the transcript is expressed in the experimental cell type and would not be observed as RBP-binding in other cell types. The second set of negative samples was generated by randomly sampling CLIP peaks of other RBPs. This ensures that any CLIP-seq biases (such as U-bias during UV-C cross-linking (38,39)) are present in both positive and negative samples and prevents the model from performing a biases-based sample discrimination during the training. Together, this yields a three-class training set, where class 1 corresponds to positive samples and class 2 and 3 correspond to negative samples. Samples of class 2 and 3 were sampled at a 3:1 ratio with respect to class 1. Finally, generated samples were randomly split into train, validation and test sets at a ratio of 70:15:15, respectively.

Pysster model

The *pysster* Python library (25) was used for implementation of the model which consists of three subsequent one-dimensional convolutional layers, each with 150 filters of size 18, followed by a single fully connected layer with 100 units. The ReLU (Rectified Linear Unit) activation function

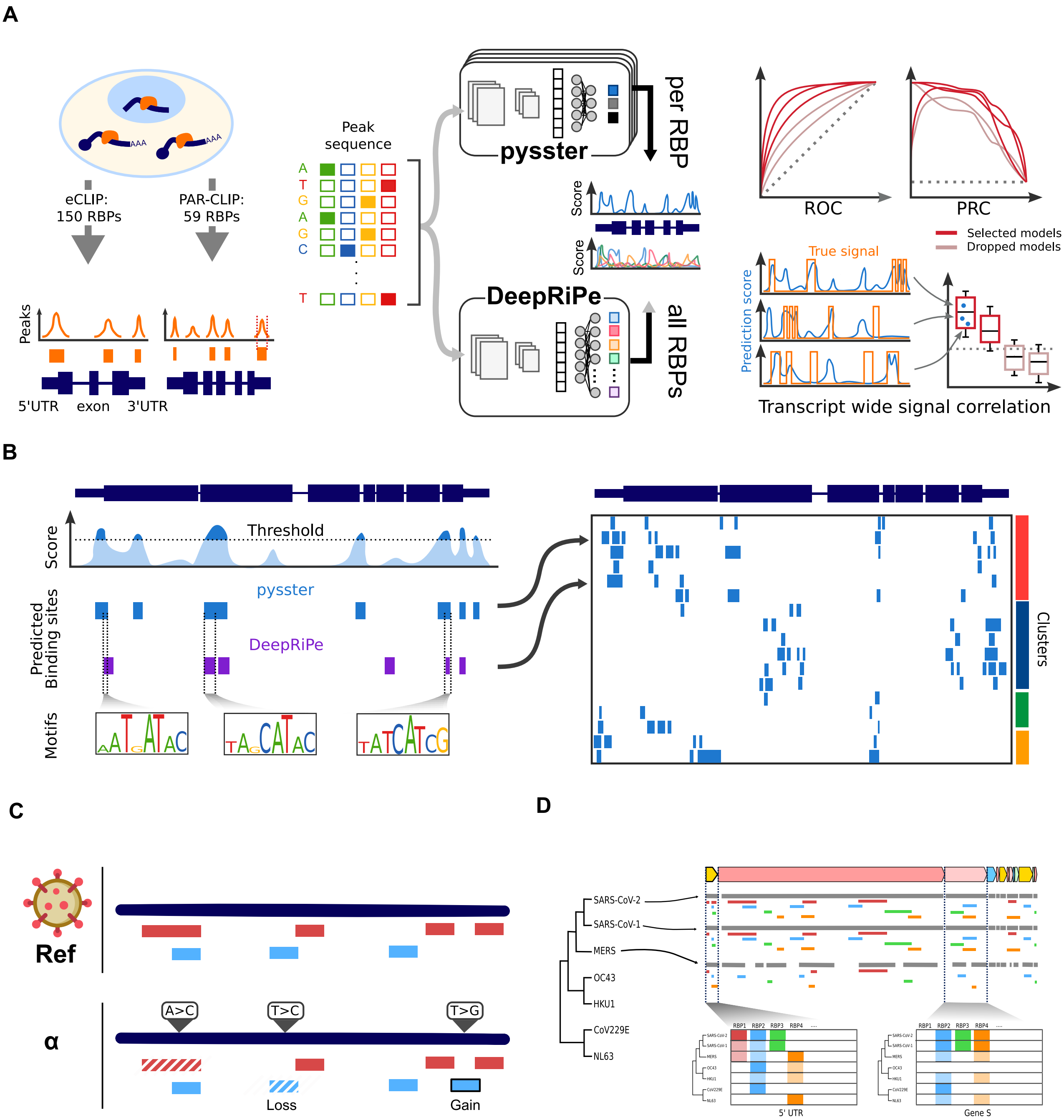


Figure 1. Pipeline of the computational mapping of the human - SARS-CoV-2 protein–RNA interactome. (**A**) (Left panel) Interactions between RNA-binding proteins (RBPs) and transcripts can be experimentally measured through eCLIP and PAR-CLIP protocols, enabling the quantification of locally accumulated reads, and the calling of peaks. Such peaks were obtained for 150 RBPs from eCLIP data (24), and for 59 RBPs from PAR-CLIP data (88). (Middle panel) Sequences from these peaks were used to train two deep learning models, composed of convolutional neural networks enabling the detection of complex sequence motifs. These models can then be applied to predict for a given sequence its potential for binding by a RBP. The pysster models are trained separately for each RBP, while DeepRiPe is trained in a multi-task fashion and simultaneously for all input RBPs. (Right panel) A selection of high-performance models was established through evaluation of performance of the models, from overall performance metrics to in-practice, sequence-wide evaluation. (**B**) All retained models were applied to scan the entire genome of SARS-CoV-2, and binding sites were predicted from consecutive, high-prediction scores positions. Sequence motifs underlying predicted RBP binding sites were also identified by interrogating both CNNs via Integrated Gradients. Predictions were compiled in the first *in silico* map of host-protein−viral RNA interactome for SARS-CoV-2. (**C**) The prediction models were applied to evaluate the impact of mutations from variants of concerns, (**D**) as well as to evaluate the evolutionary trajectory of affinity of host RBPs to other coronaviruses' genomes.

is applied to each intermediate layer output and a maximum pooling layer is added after every convolutional layer. Finally, a fully connected layer with 3 units, one for each of the three output classes, is added. Dropout (40) with a rate of 0.25 was applied to each layer, except for input and output layers. The model was trained with the Adam optimizer (41) using a batch size of 512 and a learning rate of 0.001. For each RBP, we trained for at most 500 epochs and stopped training in case the validation loss did not improve within the last 10 epochs.

Pysster binary classification threshold

As pysster models are trained as a 3-class classification problem with class imbalance, we re-calibrated each model for the binary classification task by introducing a binary decision threshold t_m on the predicted positive-class probability scores. For each model m, t_m is defined as the threshold which maximized the F1 performance of the model with respect to bound vs. unbound binary classification obtained by pooling class 2 and 3 samples into a common 'unbound' class. This threshold is used to identify bound regions in the viral sequence.

DeepRiPe model

To extend the set of RBPs explored, we obtained pre-trained DeepRiPe models from Ghanbari *et al.* (26) and retained models for 33 out of the 59 RBPs, filtering out models where no informative sequence motif could be learned by the model. The PAR-CLIP-based models used in this study are modified versions of the DeepRiPe neural network, where only the sequence module to extract features from the RNA sequence is used. Briefly, the model consists of two convolutional layers, one fully connected layer and one output layer that contains k sigmoid neurons to predict the probability of binding, one for each RBP. Each convolutional layer has a rectified linear unit (ReLU) activation, followed by a max pool layer and a dropout layer with probability of 0.25. 90 filters with length 7 and 100 filters of length 5 were used for the first and second convolution layers, respectively. The fully connected layer has 250 hidden units and a ReLU activation. Details in data preparation and model training are outlined in Ghanbari et al. (26).

Single-nucleotide predictions

The pysster and DeepRiPe positive-class prediction score corresponds to the probability that the input RNA sequence is bound by the RBP of interest. By design, this score is assigned to the entire input sequence, although RBP binding sites are much more local, usually spanning only a few nucleotides (42). To predict single-nucleotide binding site probabilities from both pysster and DeepRiPe models along a RNA sequence, we employed a one-step sliding-window approach to scan over a given RNA sequence, where the predicted positive-class probability score is assigned to the center nucleotide of the input window. In order to obtain predictions over the entire RNA sequence, the $5'$ and $3'$ sequence ends were 0-padded.

Pysster performance evaluation and model selection

As the validation loss was monitored for the purpose of early-stopping, the precision-recall (PR) and F1-score performance of the pysster models was evaluated on the test set. Models with an area under the PR curve (auPRC) of less than or equal to 0.6 were deemed poor quality and thus excluded from the downstream analysis.

Training datasets were sampled at a fixed positive-negative ratio which hardly reflects the ratio of bound and unbound sites of RNA transcripts found *in vivo*. In practice, we expect that for some transcripts regions, binding sites of a particular RBP to be not observed over several kilo-bases, while other regions, such as 5' and 3' untranslated regions (UTRs), might harbor a dense clustering of binding sites. To measure the ability of pysster models to accurately predict *de novo* RBP binding-sites along whole-length RNA transcripts, we introduced the concept of Performance-In-Practice (PIP), which measures how well the single-nucleotide prediction score of the model correlates with experimentally identified CLIP-seq peaks. For a given RNA sequence, the PIP of a model is defined as the Spearman correlation coefficient (SCC) between the truncated prediction scores p_i^{trunc} and a binary vector obtained by labeling all positions that fall within eCLIP peaks with 1 and 0 otherwise. Here, p_i^{trunc} refers to a modified version of the prediction score p_i defined as

$$p_i^{trunc} = \begin{cases} p_i, & \text{if } p_i \geq t_m \\ 0, & \text{otherwise} \end{cases}$$

where t_m is a threshold obtained for each model. For each model, we performed extensive PIP analysis on the human transcriptome as follows. First, we selected the set of transcripts which contain at least one eCLIP peak for the given RBP. From this set, we uniformly drew 100 transcripts without replacement as hold-out transcripts. Subsequently, we intersected positive and negative training samples with the hold-out transcripts and discarded all samples that overlap with any of the hold-out transcripts before retraining pysster on the remaining training samples. We used the resulting models to predict along the hold-out transcripts and compute the PIP score for each hold-out transcript. Finally, models with a median PIP score of less than or equal to 0.1 were excluded from downstream analysis.

Comparison of pysster models with ENCODE eCLIP-based DeepRiPe models

We evaluated the correlation of prediction scores from pysster and DeepRiPe models, trained on data derived from the same ENCODE eCLIP experiments, on the SARS-CoV-2 RNA. To this end, we gathered pre-trained eCLIP DeepRiPe models from (26) and intersected them with high-confidence pysster models, yielding a total overlap of 53 RBPs. Subsequently, we performed scanning (Methods) and, for each RBP, computed the correlation (PCC) of position-wise prediction scores between the pysster and DeepRiPe models.

Estimating significance of prediction scores

To directly control the false positive rate of predicted single-nucleotide binding scores from both pysster and DeepRiPe models on the viral genome, we estimated prediction score significance via a RNA sequence permutation test. In order to obtain a null-distribution of predictions (positive-class) scores, we first computed the di-nucleotide frequencies on the viral RNA. Next, we performesd frequency-weighted sampling of di-nucleotides to construct a set of $N = 10\,000$ null-distributed inputs. Null-distributed prediction scores for each model were then obtained by predicting on those sequences. A p-value is assigned to each observed prediction score p_i in the viral sequence by computing the fraction of scores from the null distribution p_j^{null} for which $p_j^{null} > p_i$, $j = 1, ..., N$.

Predicting RBP binding sites

We predicted RBP binding sites on the viral RNA sequence using predicted single-nucleotide binding scores together with estimated p-values. For each pysster model, we classified nucleotides in the viral RNA as 'bound' if the predicted probability score was equal or greater than the estimated binary threshold t_m and the score was found to be significant ($P < 0.01$). Regions with a consecutive stretch of bound nucleotides of at least length 2 were then defined as a predicted RBP binding site. Neighboring predicted binding sites that are spaced by <10 nucleotides were merged to a single predicted binding site. Note that for DeepRiPe models, nucleotides in the viral RNA were considered 'bound' if the probability score is found to be significant ($P < 0.01$) and no score threshold was applied.

Base-wise feature attribution via Integrated Gradients

To gain insight into which RNA sub-sequences are driving factors for RBP binding, we compute sequence importance scores using Integrated Gradients (IGs) (26,43). Starting from an input baseline, IG performs a step-wise linear path interpolation between the baseline and the actual input sequence and computes the gradients of the interpolated inputs with respect to an output neuron. That is, we obtained a vector of importance scores over the input sequence which indicate which nucleotides of the input contributed most toward the prediction. Here, we chose the 0-vector (i.e. the one-hot encoding of all nucleotides is set to 0) as the baseline and performed 50 baseline-input interpolation steps. To obtain sequence importance scores for a given predicted binding site, we computed IGs with respect to an input window centered around the predicted binding site. For sequence-motif construction, the heights of nucleotides in the input sequence was given by the feature attribution weights.

Analyzing mutations in variants of concern

Mutation information of 11 SARS-CoV-2 viral variants of concern (alpha, beta, delta, epsilon, eta, gamma, iota, kappa, lambda, mu, omicron) was obtained from the UCSC genome-browser for the SARS-CoV-2 virus (44), and converted into VCF format. For each variant of concern, we first created a 'mutated' genome, using the viral reference sequence and the set of variant of concern specific mutations. We then centered a window at the reference position of each mutation and extracted the mutated sequence for subsequent prediction via each model. We noted that for cases where mutations were in close proximity with each other, extracted windows might contain multiple mutations. This is crucial, as only their combination might lead to gain- or loss-of-RBP-binding. The resulting prediction score on each alternative allele (ALT) was then compared with the prediction score of the same window on the reference sequence (REF). To quantify the impact of each mutation, we compute a delta score between the prediction score of ALT and REF sequence:

$$\Delta_{score} = score_{ALT} - score_{REF}. \quad (1)$$

Mutations with a positive delta score sign represent 'gain-of-binding' (GOB) events, while mutations with negative sign represent 'loss-of-binding' (LOB) events. To further narrow down the set of mutations, we compiled a subset of mutation that lead to a gain- or loss-of-binding (GOB and LOB), defined as instances where (in case of LOB) the REF score is passing the binding score threshold and p-value while the ALT does not, or vice versa (in case of gain-of-binding).

In silico mutagenesis

While many mutations have been observed across sequenced samples during the pandemic, some mutational events have not yet occurred, and their impact on the fitness of the virus is thus unknown. To fill this gap, we performed *in silico* probing of the effects of all possible point-mutations on RBP binding across the SARS-CoV-2 genome. At each viral genome position, the reference base was mutated to each of the three alternative bases. Subsequently, prediction was performed on the input windows derived from each ALT allele using all high-quality pysster models. Finally, an impact score was computed and a set of change-of-binding mutations is compiled.

Comparative analysis of human coronaviruses

Besides SARS-CoV-2, we obtained reference sequences for 6 other human coronaviruses, including SARS-CoV-1, MERS, HCoV-229E, HCoV-HKU1, HCoV-NL63 and HCoV-OC43 from NCBI (45). Using high-quality models from both pysster and DeepRiPe, we performed single-nucleotide binding prediction along each viral RNAs. Next, we computed prediction empirical p-values for each viral sequence by generating a dedicated null distribution of scores for each virus and RBP. RBP binding sites across viruses were then predicted as described previously. We evaluated genomic-element preference across a subset of shared viral genomic locations (ORF1ab, E, N, M, S, 5′ UTR, 3′ UTR) for each RBP and virus by intersecting the predicted set of binding sites of each virus with its RefSeq annotations. To compute multiple sequence alignments (MSA) between genomic elements of coronaviruses, we used the ClustalO (46) algorithm with default parameters.

Functional annotation of RBPs

To assess the potential role of RBPs with predicted binding on viral RNA sequences, we manually curated all RNA-related functions of the 88 RBPs with good predictive models using the GeneCards, Uniprot and RBP2GO databases (47).

Public COVID-19/coronaviruses OMICS data

To assess regulatory information of RBPs across available coronavirus/COVID-19 multiOMICS data, we downloaded evidence from 22 studies. We imported study-relevant supplementary tables via knowing01 (48), which harmonizes data tables and links results to molecular information, like human gene symbols, UniProt identifier, variant positions as available in the proprietary CellMap unified data model (Version 2022/03). After loading the 88 RBP names with high-confidence models, a list of 85 Gene Symbols with associated relevant annotations was returned. Two RBPs were dropped, namely DND1 and SRRM4, due to absence of associations. In addition, L1RE1 and ORF1 matched the same gene and thus the corresponding RBP was reported only once. To ensure that all RBP human gene symbols are identically named in African Green Monkey OMICS data, we used VeroE6 cells linked to human symbols.

A total of 97 research results were gathered and organized into six groups of evidence. The first group corresponds to RBP-SARS-CoV-2 interactomes established from affinity purification and mass spectrometry (18,19,29) and computational predictions in UTRs and Spike viral regions from catRAPIDomics (49) and PRISMNet (50). The second group gathers results from viral-host protein–protein interactions (PPIs) measured by affinity-purification followed by mass spectrometry (7,28)] and yeast two hybrid screenings (51). The third group corresponds to functional evidence from multi OMICS data, including the regulation of the host proteomics, phosphoproteomics, ubiquitinomics and transcriptomics up to 24 h after coronavirus infection (7,52), as well as the effectome, which includes deregulated host proteins 72 h after SARS-CoV-2 induced expression of each of the viral proteins (7). The fourth group includes multiomics data on SARS-CoV-1, for comparison. The fifth group includes CRISPR screens which probe cell survival few days after viral infection with single genes knockouts in human (28,53–55) or African green monkey [(9)] cell lines. Finally, the sixth group gathers data related to patients : genome-wide association studies (GWAS) linking human genetic variation to COVID-19 disease severity (27), and patient OMICS data, including proteomics and transcriptomics regulation of whole blood, serum or plasma of mostly inpatients (30–37).

To retrieve RBPs from these studies, we used the adjusted P-values and other data-specific cutoffs provided by each study, whenever this information was available, in order to report only significant hits. Even if the criteria used to identify significant hits were different from one study to another, most studies, e.g. interactomes of PPI networks, usually provide a core set of identified proteins, where strict p-value cutoffs were applied (e.g. FDR $\leq$ 0.05 for both Schmidt *et al.* (19) and Flynn *et al.* (18)), as well as expanded sets where cutoffs to identify significant hits were relaxed (e.g FDR $\leq$ 0.2 for Schmidt *et al.*). For data integration, we provide both hits identified at stringent cutoffs, as well as hits identified at lax cutoffs from the corresponding studies. On the OMICs data, we report a stringent cutoff (adjusted P-value < 0.01) and a lax one (adjusted P-value < 0.1), whenever available. Few data sets only provided raw P-values (7,30,32,34,52) for which we used a lower cutoff of P-value <1e−4. Patient transcriptomics data was also used at the stringent cutoff of P-value <1e−4, due to the inflation of regulated genes on typical cutoffs. For GWAS data we employed a genome-wide (P-value < 5e−08) and nominal (P-value < 0.01) significance cutoff, for stringent and lax cutoffs, respectively. Finally, we annotated 85 RPBs with regulated molecules and visualized the number of evidences of RBPs in each data set in a count matrix.

Statistical analysis of RBP-binding on SARS-CoV-2

A permutation test was performed to identify human RBPs with a significant enrichment of predicted binding sites on the SARS-CoV-2 RNA. To this end, di-nucleotides of the SARS-CoV-2 RNA sequences were shuffled to generate 100 randomized viral sequences. Subsequently, binding site prediction was performed on each randomized sequence for each high-confidence human RBP model and the numbers of predicted binding sites were recorded. Using the thus generated null distribution of predicted binding site counts, a P-value was computed for each RBP as the number of times a randomized sequence harbored more predicted binding sites than the true SARS-CoV-2 RNA sequence, divided by the number of randomized sequences (100). A pseudo-count was added to prevent P-values of 0. Finally, we corrected for multiple testing by adjusting P-values following the Benjamini-Hochberg procedure and significantly enriched RBPs were selected using a FDR threshold of 0.1.

Statistical analysis of SECReTE-Motifs

To estimate whether clusters of RBPs show a significant enrichment of binding sites overlapping with SECReTE motifs on the SARS-CoV-2 RNA, we performed a Fisher exact test of each cluster. Specifically, a contingency table is constructed with (i) number of cluster binding sites overlapping with SECReTE motifs, (ii) number of non-cluster binding sites overlapping with SECReTE motifs, (iii) number of cluster binding sites not overlapping with SECReTE motifs and (iv) number of non-cluster binding sites not overlapping with SECReTE motifs. To obtain p-values, we took the sum over the upper tail of the hyper-geometric distribution. Finally, we accounted for multiple testing by adjuting P-values with the Benjamini-Hochberg correction.

RESULTS

Accurate model predictions in human and viral sequences

The trained pysster models showed a robust area-under-precision-recall-curve (auPRC) performance, with a median auPRC of 0.6 across all 150 trained models (Figure 2A, Supplementary Table S1). As models were used for scanning of the full-length viral genome (rather than classification of standalone examples), we further evaluate the model performance by computing the correlation of the predicted positive-class probabilities with observed ENCODE peaks on a hold-out set of human transcripts. Nearly all models showed a significant positive correlation, with a mean median Spearman correlation coefficient (SCC) across transcripts of 0.149 and a maximum median SCC of 0.38 (Figure 2B, Supplementary Table S1). Considering that eCLIP experiments have a relatively low signal correlation baseline between replicates (56), this indicates that the trained models are well-suited for the task of scanning across the viral genome. Exemplary prediction tracks for two held-out human transcripts using pysster models of QKI and TARDBP are shown in Figure 2C. In general, we observe that models which perform well with respect to the auPRC score tend to perform well in the context of RNA sequence scanning (Figure 2D). To ensure that downstream analyses are based on a high-quality set of binding site predictions, models with a median SCC of < 0.1 or an auPRC of < 0.6 were discarded. A total of 63 high-quality pysster models were thus kept for predicting on the SARS-CoV-2 genome. For DeepRiPe, we relied on the results from (26) and retained only those models where informative sequence motifs were learned during training, leaving a total of 33 RBP models for predicting on the SARS-CoV-2 genome. Of those, we selected only models for RBPs not contained in the ENCODE database, leading to the addition of 24 high-quality DeepRiPe models.

The lack of eCLIP data in a SARS-CoV-2 infection context prevents a large-scale performance evaluation of trained pysster and DeepRiPe models directly on the SARS-CoV-2 RNA sequence. Nevertheless, we obtained eCLIP data for two proteins, CNBP and LARP1, from Schmidt *et al.* (19), allowing us to partially evaluate the validity of our cross-species (human/virus) training and prediction approach. After generating training samples from CNBP and LARP1 eCLIP peaks within human transcripts (processed in the same manner as eCLIP peaks from the ENCODE experiments), we trained pysster models for both RBPs (Supplementary figure 1A). We then performed prediction along the SARS-CoV-2 RNA sequence and compared the resulting single-nucleotide prediction scores with observed eCLIP peaks as well as the eCLIP signal provided by the authors (Figure 2E and 2F, Supplementary Figure 1C). Predictions showed an auPRC of 0.72 and 0.56, respectively, and a strong correlation with the raw eCLIP signal (SCC = 0.332, P-value $< 1e{-}16$ for CNBP and SCC = 0.133, P-value = $7.96e{-}12$ for LARP1). In addition, we observed an accumulation of high-scoring positions at the location of experimentally identified peaks for CNBP and (to lesser extent) LARP1 (Figure 2F, Supplementary Figure 1C). We therefore excluded the LARP1 model from the final set of models, as it does not meet the established performance criteria (Methods), while the CNBP model was retained. Although it is difficult to assess to what extend these results generalize to other RBPs, high performing models selected in the previous section are likely suitable for cross-species, *in silico* prediction of RBP binding sites on SARS-CoV-2.

A comprehensive *in silico* binding map of human RBPs on SARS-CoV-2

We performed *in silico* binding site prediction by identifying consecutive significant and high-scoring positions within the SARS-CoV-2 genome with both pysster and DeepRiPe high-confidence models (Methods). In the following, we first demonstrate that our model makes predictions on the basis of genuine sequence features and subsequently build a computational map of predicted SARS-CoV-2-human RBP interactions. We then evaluate the enrichment of different RBPs for different viral genomic regions, as well as their putative regulatory function in the context of SARS-CoV-2 infection.

Predicted RBP binding sites coincide with known binding motifs. While deep neural networks yield high performance on the task of protein–RNA interaction prediction, they are often considered to be black-box predictors, with predictions generally difficult to interpret. Nevertheless, it is imperative to assert that the model's binding predictions are performed due to presence of genuine RNA sequence features associated with protein–RNA interaction, rather than spurious signal incorporated into the model due to technical biases in the training set. As only a fraction of RBPs are known to bind to distinct primary motifs, we performed feature importance analysis for selected RBPs in order to assess whether the sequence features underlying predictions at predicted binding sites correspond to the binding site preferences of those proteins reported in literature.

Figure 3A and 3B each show single-nucleotide resolution prediction scores of the human RBPs RBFOX2 and TARDBP, obtained from pysster models, and MBNL1 and QKI, obtained from DeepRiPe models. Predicted binding sites are shown below the prediction score tracks. We centered input windows around predicted binding sites of RBFOX2, TARDBP, MBNL1 and QKI on SARS-CoV-2 to identify individual nucleotides that were most predictive for classifying the input sequence as 'bound' (Figure 3A and 3B; bottom track). We observed that feature importance maps around predicted binding sites corresponded to known binding motifs. For instance, we observe the well-known consensus sequence (T)GCATG recognized by the splicing factor RBFOX2 (57) in the corresponding feature importance maps (Figure 3A, left), as well as the TG-repeat motif, corresponding to the sequence preference of TARDBP (58), coinciding with its predicted binding sites (Figure 3A, right). Similarly, DeepRiPe attribution maps with respect to binding sites of QKI show the canonical binding motif TACTAA(C) (59) (Figure 3B, left). Lastly, the attribution maps computed at each predicted binding site of the splicing factor MBNL1 all harbor occurrences of the characteristic YGCY motif (60) (Figure 3B, right).

Binding site predictions are robust across different datasets and prediction tools. To evaluate the robustness of viral binding site predictions across pysster and DeepRiPe, we compared predictions for a small set of RBPs where both eCLIP data (used to train pysster models) and PAR-CLIP data (used for the training of DeepRiPe models) were available. (For a comparison with DeepRiPe models trained on eCLIP data, see Supplementary Table S2.)

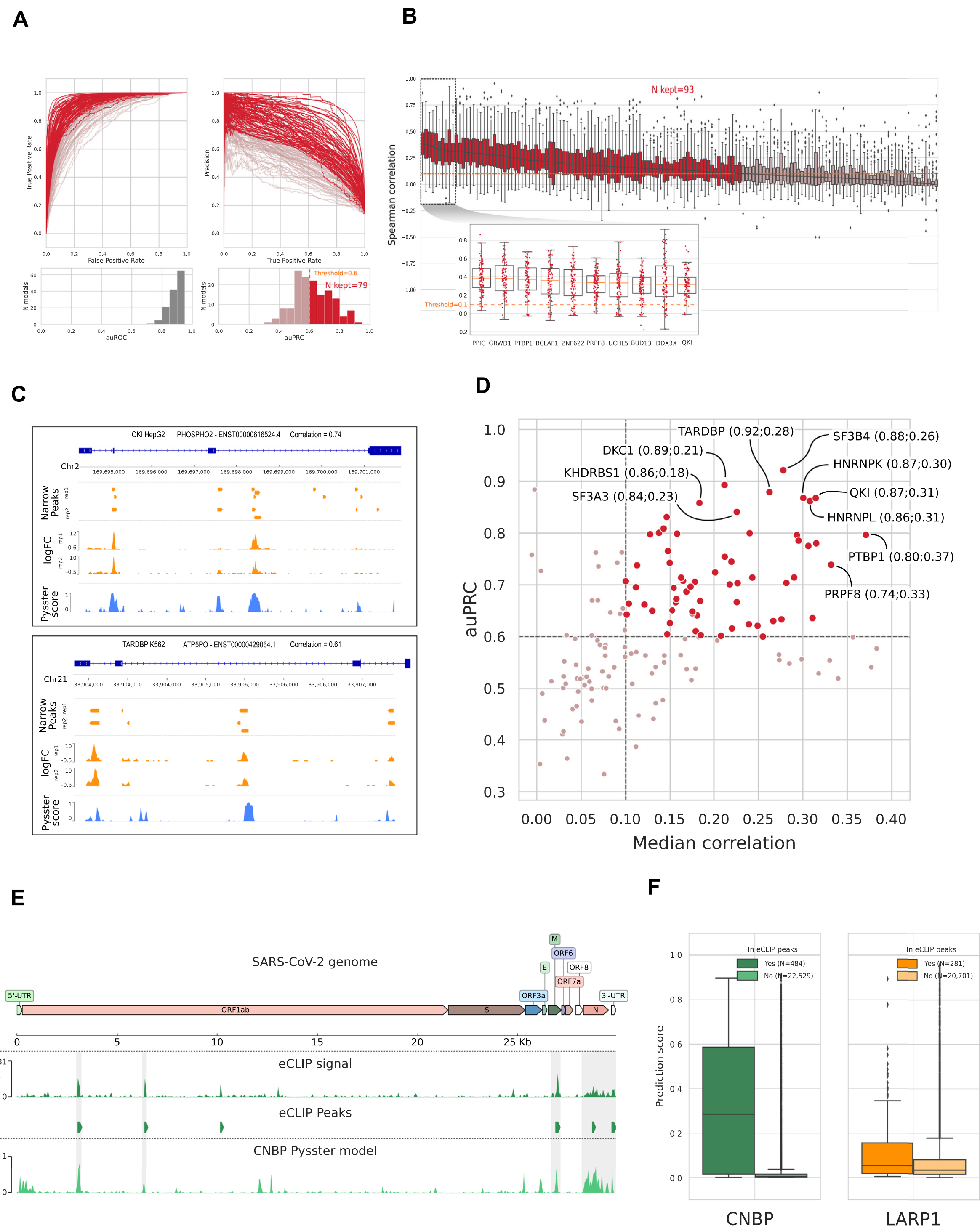


Figure 2. Evaluation of pysster models' performance and high-quality model selection. (**A**) Receiver Operating Curve (ROC) and Precision Recall Curve (PRC) for all 150 pysster models trained from ENCODE eCLIP datasets. A first threshold of 0.6 was set on the area under the PRCs (auPRC), leading to a subset of 79 models passing the threshold. (**B**) Boxplots of correlations between eCLIP and prediction scores from 100 left-out transcripts per RBP model. This correlation highlights the performance of models in a realistic context of full-sequence-length scan. A second threshold was thus set on the median correlation coefficient, leading to a subset of 93 models passing the threshold. The 10 models with highest median correlation are displayed in a detailed sub-plot. (**C**) Genome-browser view illustrating the comparison between eCLIP signals and model prediction scores over full-length transcripts. Two of the best models are presented, with signal from left-out transcripts with high correlation between eCLIP log-fold-change signals and prediction scores from the pysster models. (**D**) Scatterplot of the AUPRC and median correlation values for each model, highlighting the final subset of high-quality models. The top 10 models are labeled. (**E**) Comparison of genome-wide eCLIP signal and pysster prediction scores from the CNBP eCLIP datasets generated over the SARS-CoV-2 genome by (19). (**F**) Boxplot of pysster prediction scores from position within or without overlap from called narrow peaks, for the CNBP model and the LARP1 model (*t*-test *P*-values: $< 1e{-}16$ for CNBP; $2.44e{-}6$ for LARP1).

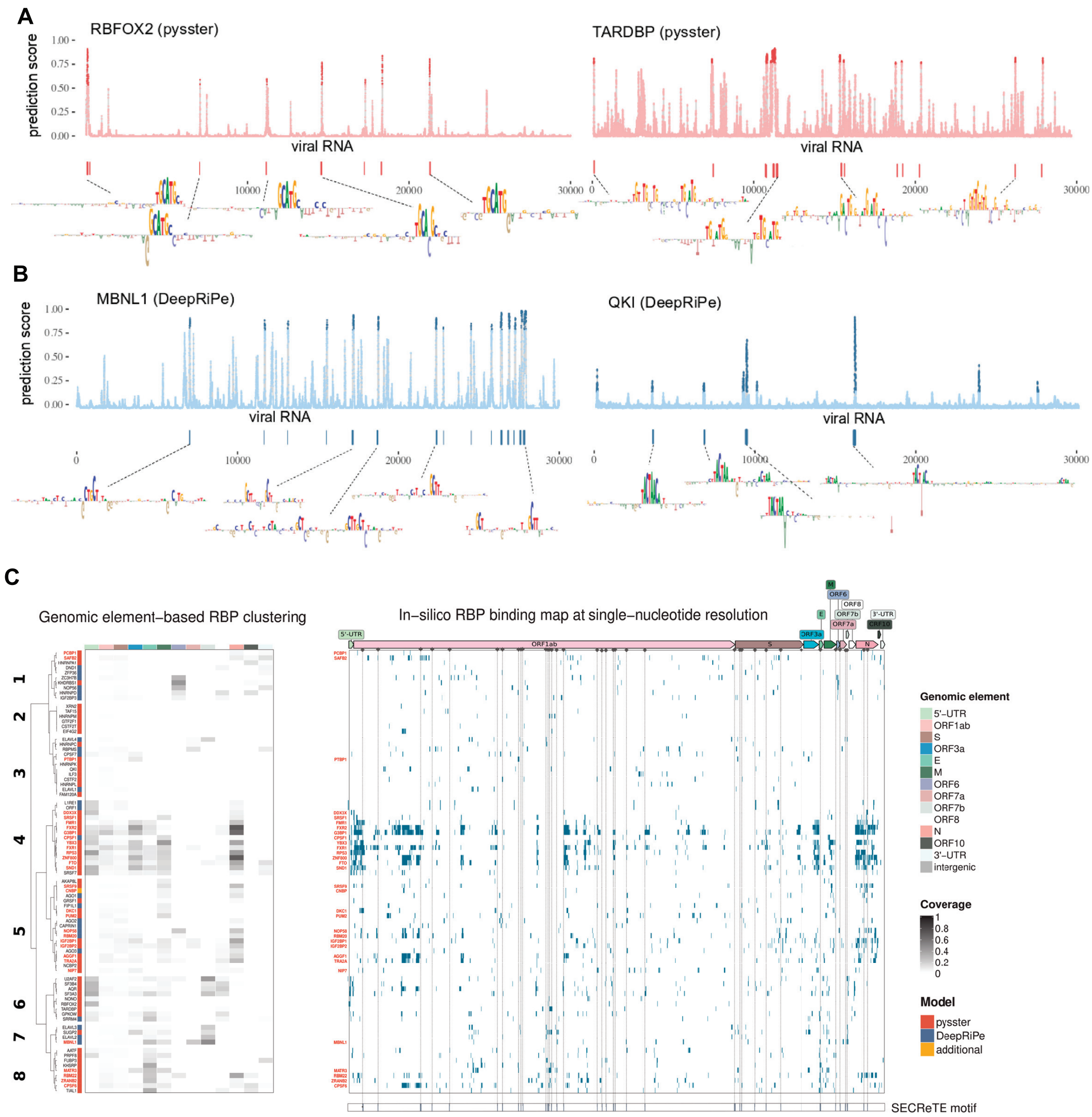


Figure 3. Computational map of RBP binding on SARS-CoV-2. (**A**) Single-nucleotide probability score for RBFOX2 (left) and TARDBP (right) RBP binding as computed by the corresponding pysster models across the whole SARS-CoV-2 genome. The higher the score, the higher the likelihood of a binding event at that position. Points highlighted in strong color correspond to significant predictions, i.e. with bound probability significantly higher than random (empirical p-value < 0.01, see Methods). Wider predicted binding sites, encompassing more than one significant position are shown as vertical bars underneath each prediction profile, together with their corresponding binding motifs as extracted by means of attribution maps (see Methods). (**B**) Single-nucleotide probability score for MBNL1 (left) and QKI (right) RBP binding as predicted by the corresponding DeepRiPe models. Significant positions (empirical *p*-value < 0.01) are highlighted in strong color, and predicted binding sites together with their corresponding motifs are shown underneath. (**C**) Clustering of RBPs based on predicted binding site coverage of genomic annotations of SARS-CoV-2 for both pysster and DeepRiPe RBPs (left panel). *In silico* RBP binding map, at single-nucleotide resolution, for both pysster and DeepRiPe RBPs (right panel). RBP names in bold and red indicate those with statistically significant number of predicted binding sites when compared against predictions in 100 shuffled genomes, see methods. SARS-CoV-2 SECReTE motifs from (67) are shown below, with vertical lines helping the visualization of overlaps with predicted binding sites.

Among a total of 20 overlapping RBPs, 12 were contained in the sets of high-quality models for pysster and DeepRiPe, namely TARDBP, CSTF2, IGF2BP1, PUM2, CSTF2T, QKI, IGF2BP2, IGF2BP3, CPSF6, FXR1, FXR2 and EWSR1. For each of the 12 RBPs, we then computed the correlation between the pysster and DeepRiPe prediction scores across single-nucleotide positions on the viral genome. We observed a signal correlation higher than 0.1 for 8 out of the 12 RBPs, with a correlation coefficient ranging from a maximum of 0.64 (TARDBP) to a minimum of 0.15 (CPSF6) (Supplementary Table S3). In general, we observed a higher overlap between pysster and DeepRiPE binding site predictions for RBPs harbouring well-defined RNA sequence motifs, such as QKI, TARDBP, PUM2, CSTF2, and to a less extent, FXR1-2 and IGF2BP1-3. In addition, feature attributions maps at overlapping predicted binding sites of pysster and DeepRiPe with respect to QKI and TARDBP (Supplementary Figure S2), highlight the presence of the known binding motifs for these two RBPs.

Binding preferences and clusters of human RBP predicted sites on the SARS-CoV-2 genome. Given the set of 88 high confidence models comprised of pysster models trained on ENCODE eCLIP data and supplemented by DeepRiPe models trained on PAR-CLIP data, we build a comprehensive *in silico* SARS-CoV-2 / human RBP binding map. Note that eCLIP DeepRiPe models were not included here, as these models were trained with an architecture optimized for PAR-CLIP datasets (26), while in addition, high predictions correlation was observed between pysster and DeepRiPe eCLIP models for many RBPs (Supplementary Table S2).

Figure 3C (right) depicts the predicted binding profiles of 84 (out of 88) human RBPs which harbor at least one predicted binding site on the SARS-CoV-2 sequence. Among those, 31 were found to exhibit a significant enrichment of predicted binding sites on SARS-CoV-2 compared to a background of randomly shuffled sequence of the same length (see Materials and Methods, Figure 3C, Supplementary Table S4). Next RBPs were clustered into eight classes based on their relative predicted binding site coverage across different genomic regions of the SARS-CoV-2 genome (Figure 3C, left). While some clusters of proteins exhibit sparse predicted binding signal across the SARS-CoV-2 genome (such as clusters 2 and 3), other clusters contain almost exclusively RBPs with enriched predicted binding across the whole SARS-CoV-2 genome (cluster 4).

We observe overall extensive RBP binding coverage mostly at 5' UTRs and genomic regions coding for E, M and N structural proteins, and less coverage at the spike S gene, as well as the viral 3' UTR. Clustering of predicted binding sites groups together RBPs with similar roles in RNA processing and viral regulation, RNA recognition mechanisms and preferences for certain genomic locations. Cluster 4 showed the highest accumulation of RBPs with predicted binding enriched on the SARS-CoV-2 sequence. This clusters corresponds to a group of well-known regulators of RNA processing, including proteins from the FXR family (FXR1, FXR2 and FMR1) (23), which extensively bind the viral 5' UTR, as well as the ORF1ab and subgenomic RNAs. Other proteins in this cluster include DDX3X, regulator of RNA translation and host target against SARS-CoV-2 infection (61), splicing regulators (SR) SRSF1 and SRSF2 and RNA demethylase FTO, implicated in HIV infection (62,63). Further, cluster 1 predominantly harbors RBPs with predicted binding preference for the viral 3' UTR, including regulators of RNA stability and proteins involved in 3' end formation and/or regulation of translation. Cluster 6 is comprised of RBPs which preferentially bind to the 5' UTR of SARS-CoV-2 and are involved in splicing (23). It contains NONO, a member of the paraspeckle complex, previously found in the RBP interactome of SINV infected cells (13), as well as TARDBP, a protein that localizes to P-bodies and stress granules and was shown to bind to the 5' UTR of SARS-CoV-2 in a recent study (64). Cluster 5 includes RBPs with diverse functions who preferentially bind to the N and M viral genomic regions, while RBPs in cluster 7 and 8 were mostly predicted to bind ORF7b, as well as E and M regions. Besides the splicing regulators MBNL1 and sUGP2, cluster 7 contains members of the ELAVL family, previously shown to be prone to be hijacked during viral infections (65). While most RBPs in cluster 8 were not found to be functionally related in literature, RBPs KHSRP and MATR3 have been shown to act as restriction factors in SINV infection (66).

Predicted RBP binding sites overlap with SECReTE motifs. Haimovich et al. (67) recently identified the presence of a unique *cis*-acting RNA element, termed 'SECReTE' motif, which consists of 10 or more consecutive triplet repeats, with a C or a U present at every third base, on the sequences of both (−) and (+)ssRNA viruses. In context of SARS-CoV-2, a total of 40 SECReTE motifs have been identified in the viral genome, with a total length of ∼1.3 kb. This motif has been suggested to be important for efficient translation and secretion of membrane or ER-associated secreted viral proteins, as well as for viral replication centers (VRCs) formation. To investigate whether predicted binding sites coincide with SECReTE motifs, we obtained exact locations of all SARS-CoV-2 SECReTE motifs from (67), and subsequently intersected them with predicted RBP binding sites of all 84 high-quality models containing at least one predicted binding site in SARS-CoV-2 (Supplementary Table S5). We observed that a total of 61 RBPs (out of 84) have predicted binding sites overlapping with SECReTE motifs. Further, 30 RBPs with at least 10% of their predicted binding sites overlapping with SECReTE motifs were identified. We next investigated whether clusters of RBPs binding to SARS-CoV-2 (Figure 3C) are enriched in binding sites overlapping with SECReTE-motifs. We find that clusters 7 and 8 show a significant enrichment (adj.P < 1.56e−05 and adj.P < 0.0667) of binding sites overlapping with SECReTE-motifs on SARS-CoV-2 genome, while clusters 1 and 5 show a significant depletion (adj.P < 0.0291 and adj.P < 0.0404) at a significance level of 0.1. For instance, cluster 8 harbors 4 (out of 9) SECReTE-associated RBPs (FUBP3, KHSRP, MATR3 and CPSF6), 3 of which (FUBP3, KHSRP, MATR3) have 25% or more of their predicted binding sites overlapping with SECReTE motifs. KHSRP is an essential RBP involved in RNA localization, RNA stability and translation, while METR3 is a regulator of RNA stability. Interestingly, most of these factors have been previously associated to viral RNA regulation (23). Similarly, all 4 RBPs in cluster 7 (ELAVL2, ELAVL3, SUGP2 and MBNL1) have > 25% of their respective predicted binding sites overlapping genomic regions harboring SECReTE motifs.

SARS-CoV-2 variants of concern show gain- and loss-of-binding events

Multiple waves of SARS-CoV-2 infections have spread across the globe, some of which resulted in the emergence of specific lineages of viral variants. The systematic sequencing of thousands of samples from infected patients enabled the description and categorization of the detected viral sequences, identifying numerous mutations in their sequence when compared to the initial SARS-CoV-2 reference genome. Some of the thus described strains have been experimentally characterized as more efficient than others, explaining in part their successful spread at local or global geographic scales (68–70). These strains have been defined by the World Health Organization as variants of concern, with 'evidence for increased transmissibility, virulence, and/or decreased diagnostic, therapeutic, or vaccine efficacy' (71). Specific subsets of mutations have been associated with each variant of concern, when mutations were represented in a majority of sequenced samples of their lineage. Notably, a special focus has been given with regards to the impact of mutations occurring within the spike-encoding gene (72), owing its importance in the initial steps of viral infection and its potential for vaccine neutralization (73). However, due to a lack of appropriate methods, the impact of these mutations at the regulatory level, such as their impact on protein–RNA interactions, has so far been largely ignored. To fill this gap, we systematically investigated the impact of observed mutations in viral variants of concern on the predicted binding of RBPs, in order to uncover potential viral hijacking of host proteins directly at the RNA level.

A catalog of high-impacting mutations across 11 variants of concern. We compiled a total of 290 mutations (193 unique mutations, 37 shared across strains) across 11 variants of concern, including alpha, delta, and omicron strains. For each variant and RBP, we evaluated the impact of the variant in terms of gain- or loss-of-binding by comparing the predicted binding probability of the reference and alternative allele. Using pysster and DeepRiPe models across 87 RBPs, we obtained a total of 25,230 impact scores, one for each pair of variant and RBP. Notably, three variants (3,037C>T, 14,408C>T and 23,403A>G) are consistently found across all viral strains, and their highest absolute delta scores were respectively associated to FTO (avg. decrease from 0.474 to 0.356), AQR (avg. decrease from 0.191 to 0.036), and NONO (avg. increase from 0.086 to 0.340). In order to prioritize pairs of variants and RBPs that show a gain- or loss-of-binding, we select a subset of pairs for which either the reference *or* alternative allele pass our binding thresholds. Note that this filter applies a XOR operation, i.e. we are interested in events that lead to either gain- or loss-of-binding (GOB, LOB). Overall, a total of 315 GOB or LOB events passed the above filter and are depicted in Figure 4A. The majority of variants introduced small delta in prediction scores, with less than 20% (61) of absolute delta-scores above 0.233 (Figure 4A). As shown in the Supplementary Figure S3A, the top 20% highest-impact variants from Figure 4A accumulate in different genomic annotations over the SARS-CoV-2 genome. Interestingly, among the RBPs impacted by these mutations, we find that some variants of concern present multiple high delta score mutations for SRSF7 (delta, kappa) and YBX3 (lambda), as well as L1RE1, RBPMS, SND1, ZRANB2 (omicron) (Supplementary Figure S3B and S3C). Additionally, the omicron variant of concern harbors a particularly large number of variants predicted to impact binding of ORF1 protein (from LINE-1 retrotransposable element).

Systematic point-wise in silico *mutagenesis reveals hypothetical high-impact variants.* New viral strains are continuously emerging, some of which are characterized by a faster spread due to newly acquired sequence variants, highlighting the importance of a continuous monitoring of viral variants which may result in a selective advantage on the protein or RNA regulatory level. To anticipate and quantify the impact of potentially unobserved variants, we perform a systematic *in silico* mutagenesis by generating all possible point mutations across the SARS-CoV-2 genome and score each hypothetical mutation with respect to its impact on RBP binding. Figure 4D and 4E show exemplary *in silico* mutation tracks for PUM2 and FTO, respectively, with observed reference prediction scores depicted at the top and the impact of gain- and loss-of-binding variants shown at the bottom. Note that for visualization purpose, only the delta score of the alternative allele with the highest impact is shown for each position and RBP. Supplementary Figure S4 shows an impact catalogue of 29,903 × 63 single-nucleotide variants across all SARS-CoV-2 genome positions and 63 pysster models. Complete set of hypothetical variants together with their impact scores is available at https://sc2rbpmap.helmholtz-muenchen.de/.

High-impact sequence variants disrupt known RBP-binding motifs. As *in vivo* RBP-binding is usually driven via the recognition of short sequence motifs, we investigated whether high-impact variants cause gain- or loss- of known binding motifs. To this end, we gathered from each strain the top 10 variants with highest absolute delta scores, as illustrated in Figure 4B and 4C for strains alpha and delta, respectively. This represented a total of 69 unique mutation-RBP pairs, 19 of which were found in more than one strain. As expected, the majority (54/69) of their delta scores is found to be in the top 1% of the distributions from the *in silico* mutagenesis. We then computed feature attribution scores, centered at the position of each high-impact variant. Feature attribution maps for the subset of candidate high-impact variants of the alpha and delta strain are shown in Figure 4B and 4C, respectively. Indeed, we observe that variants with high negative delta score tend to disrupt known binding motifs of human RBPs. For instance, transition T>G at position 22,917, as seen in the delta strain (Figure 4C) (as well as in top mutations from epsilon and kappa strains) decreases the prediction score for PUM2 from 0.795 to 0.158, with only 0.0015% in silico variants showing a

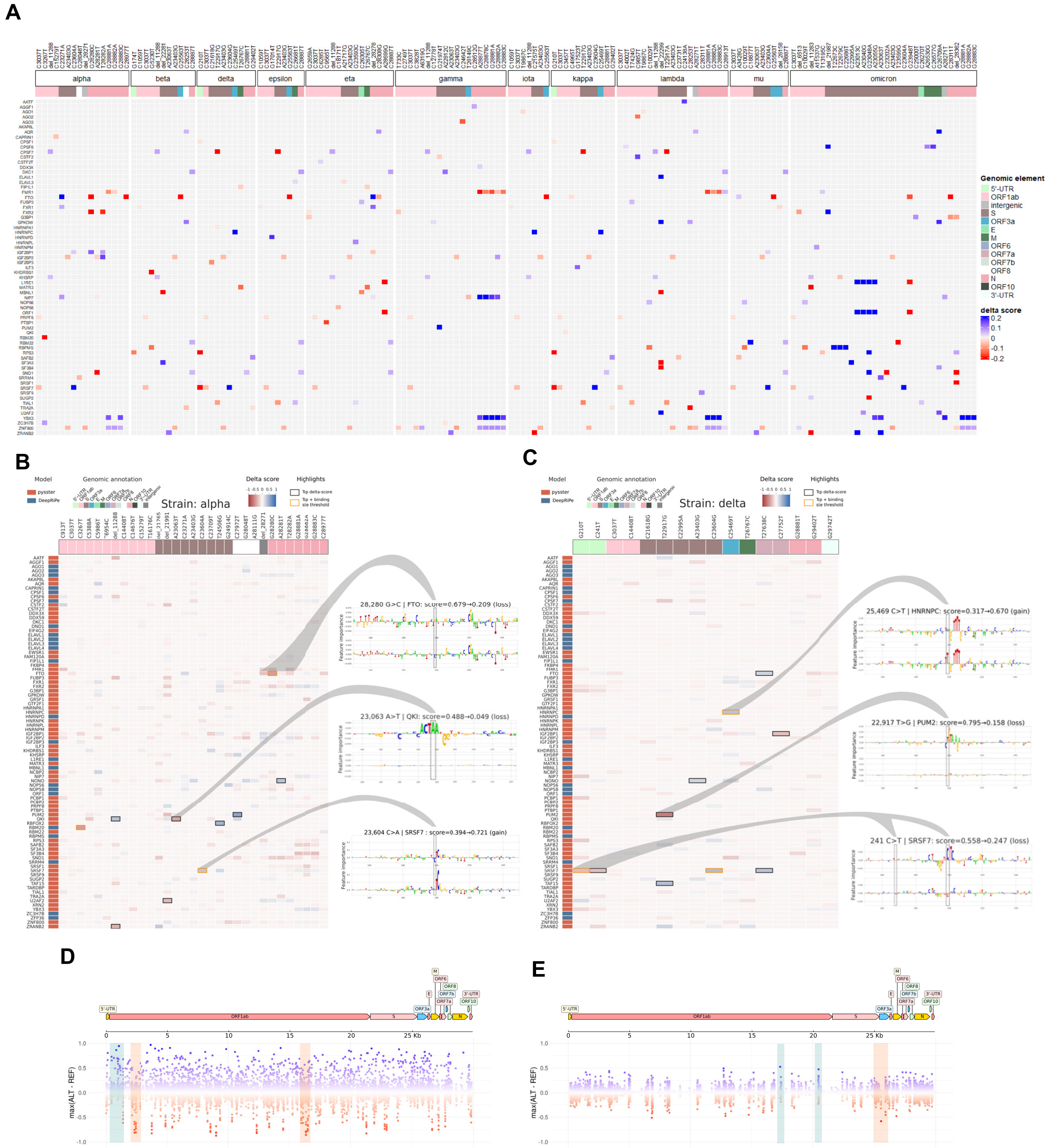


Figure 4. Impact of mutations from SARS-CoV-2 variants of concern on predicted binding sites. (**A**) Joint heatmap of delta scores from the 290 identified mutations in the different SARS-CoV-2 variants of concern. Delta scores represent the difference in prediction score of a prediction model between alternative and reference sequences centered on each mutation. Only the 315 impacts labeled as change-of-binding are colored. Delta score color scale is capped so as to show low delta score impacts. RBPs and mutations without any such impact across strains are dropped from the heatmap. (**B**) Complete heatmap of delta scores from 31 mutations associated to the alpha viral variant. The top 10 with highest absolute delta scores are lined out, with yellow color indicating the ones labeled as change-of-binding. Some sites are further investigated through integrated gradients, comparing the sequence motifs identified by the prediction models against known motifs from mCrossBase (89). (**C**) Complete heatmap of delta scores from 16 mutations associated to the delta viral variant. (**D**, **E**) Results from the *in silico* mutagenesis over the SARS-CoV-2 genome for PUM2 and FTO, respectively. Nucleotides across the viral genome were perturbed towards the three alternative bases, generating a reference distribution of possible delta scores, notably highlighting positions with highest impacts. Highlighted regions show hypothetical variants that are predicted to lead strong gain- (blue) or loss- (red) of-RBP-binding.

Table 1. Subset of high delta score mutations passing binding sites thresholds

	RBP	Variant	Strain	Genomic element	REF score	ALT score	delta score	Impact
0	SRSF7	G210T	delta, kappa	5' UTR	0.768	0.457	− 0.311	loss
1	RBM20	C3267T	alpha	ORF1ab	0.813	0.336	− 0.477	loss
2	RBM22	C18877T	mu	ORF1ab	0.338	0.614	0.276	gain
3	HNRNPC	C21575T	iota	S	0.374	0.840	0.467	gain
4	MBNL1	del_22281	beta	S	0.800	0.006	− 0.795	loss
5	ELAVL1	del_22299	lambda	S	0.070	0.632	0.562	gain
6	SF3B4	del_22299	lambda	S	0.871	0.128	− 0.744	loss
7	SF3A3	del_22299	lambda	S	0.860	0.273	− 0.587	loss
8	U2AF2	del_22299	lambda	S	0.543	0.980	0.438	gain
9	GPKOW	del_22299	lambda	S	0.297	0.841	0.544	gain
10	MBNL1	del_22299	lambda	S	0.803	0.398	− 0.405	loss
11	SF3A3	C22995A	omicron	S	0.081	0.808	0.726	gain
12	ORF1	A23013C	omicron	S	0.014	0.621	0.608	gain
13	ORF1	A23040G	omicron	S	0.006	0.673	0.666	gain
14	ORF1	G23048A	omicron	S	0.006	0.606	0.600	gain
15	SND1	G23048A	omicron	S	0.187	0.791	0.604	gain
16	SRSF7	C23604A	alpha, mu	S	0.394	0.719	0.326	gain
17	SRSF7	C23604G	delta, kappa	S	0.394	0.792	0.398	gain
18	HNRNPC	C25469T	delta, kappa	ORF3a	0.317	0.670	0.352	gain
19	FTO	G25563T	beta, epsilon, iota, mu	ORF3a	0.633	0.080	− 0.552	loss
20	FTO	del_28278	eta	N	0.335	0.683	0.348	gain
21	FTO	G28280C	alpha	N	0.679	0.209	− 0.470	loss
22	ORF1	A28699G	eta	N	0.597	0.141	− 0.456	loss

lower delta score. As is clearly visible from the feature attribution analysis (Figure 4C; middle-right), the variant disrupts the well-known PUM2 binding motif TGTATAT. In a similar manner, transversion A>T at position 23,063 from the alpha strain (Figure 4B; also found in top mutations from beta, gamma, and mu strains) decreases the prediction score for QKI from 0.488 to 0.049, with 0.006% *in silico* mutations show a low delta score. Here, the feature attribution profiles clearly highlight how the known QKI binding motif ACTAA was detected by the model in the reference sequence, and how the mutation leads to a loss of this motif. Lastly, the transversion G>C at position 28,280 in the alpha strain (Figure 4B) decreases the prediction score for FTO binding from 0.679 to 0.209, and only 6 (0.00007%) *in silico* mutations show a delta score lower than the one observed (Figure 4D). Although no clear motif is found within the window, the heights of the nucleotides at the position of the mutation are reduced compared to the reference sequence, reflecting the decreased prediction score.

High-impact gain- and loss-of-binding events across viral strains. Among the above set of top 10 highest impact variants per viral strain, we select those that conform to strict gain- or loss-of-binding. We identify a total of 23 (out of 69) change of binding events across 17 variants and 13 RBPs (Table 1). The first example corresponds to a transversion G>T at position 210 in the 5'UTR from the delta and kappa strains, predicted to induce a loss-of-binding for SRSF7, which we had confirmed from the loss of binding motif (delta strain heatmap, see Figure 4C). Further, from the ORF1ab gene, two examples of a loss of binding for RBM20 by the C>T transition at position 3,267 (strain alpha), and a gain-of-binding of RBM22 from a C>T transition at position 18,877 (strain mu). From the S gene, a gain-of-binding is reported for HNRNPC, induced by a C>T transition at position 21,575 (strain iota), in addition to another gain-of-binding reported for SF3A3, from a C>A transversion at position 22,995 (strain omicron). Two mutations occurring in the ORF3a gene are passing our filters for two RBP impacts: the transition C>T at position 25,469 induces a gain of binding for HNRNPC in delta and kappa strains, while the G>T transversion at position 25,563 induces a loss of binding for FTO in strains beta, epsilon, iota and mu. Finally, in the N gene, we report three mutations, two of them impacting FTO binding (one gain in the eta strain, from a deletion at position 28,278; one loss in the alpha strain, from a G>C transversion at position 28,280), and a loss-of-binding of ORF1 protein (from LINE-1 retrotransposable element) in the eta strain, from a A>G transversion at position 28,699.

Individual variants impact binding of several RBPs. Among variants that surpass binding sites thresholds and lead to either gain- or loss-of-binding, several variants impact RBP binding of multiple RBPs simultaneously. For instance, a deletion at position 22,299 (S gene) identified in the lambda strain, is predicted to induce a gain-of-binding for ELAVL1, U2AF2 and GPKOW, while inducing a loss-of-binding for SF3B4, SF3A3 and MBNL1. Interestingly, all these factors are associated with splicing. Notably, the MBNL1 loss is also detected in the beta strain, through a deletion happening in a close-by location (at position 22,281, S gene), suggesting those two mutations may have been retained due to beneficial induction of similar changes in binding patterns. Another mutation which impacts multiple RBPs is the transition G>A at position 23,048 (S gene) from the omicron strain, predicted to induce binding of the ORF1 protein from LINE-1 retrotransposable element, as well as of SND1. Comparably to the MBNL1 impact, two close-by mutations from omicron were associated with a gain of ORF1 binding (transversion A>C at position 23,013, and transition A>G at position

23,040), further suggesting joint impact of these mutations on ORF1p binding. The last case of mutations with impact on multiple RBPs concerns a set of 2 mutations: C>A transversion and C>G transversion at position 23,604, in the S gene. The first is found in alpha and mu strains, while the second is found in the delta and kappa strains. Both mutations are predicted to induce a gain of SRSF7 binding, which is visualized for the alpha strain on Figure 4B through feature attribution maps.

Predicted RBP-binding across human coronaviruses

While evaluation of impact for reported variants enables the monitoring of potentially functional changes in the SARS-CoV-2 genome, evaluating changes in binding sites at longer evolutionary time scale might highlight more fundamental properties of the SARS-CoV-2 virus, as compared to other RNA viruses infecting human. We investigated to which extent predicted binding sites of human RBPs are conserved across related human coronaviruses. For this purpose, we obtained genomes and genomic annotations of 6 SARS-CoV-2-related human coronaviruses, namely SARS-CoV-1, MERS, HCoV-OC43, HCoV-NL63, HCoV-HKU1, HCoV-229E. Binding sites were predicted in analogy to SARS-CoV-2 across each viral genome using 87 high-confidence pysster and DeepRiPe models. Figure 5A shows the general predicted binding propensity of RBPs across viral genomes of the 7 coronaviruses. Overall, predicted RBP binding is conserved across coronaviruses, with the highly pathogenic viruses (SARS-CoV-1, SARS-CoV-2 and MERS) showing a highly similar predicted pattern. Further, a total of 86 (out of 87) RBPs (except FKBP4) were predicted to harbor a predicted binding site in at least one coronavirus, with only a small variability in the total number of predicted binding RBPs between individual viruses. However, we observe a greater variability of predicted RBP binding within shared genomic regions across coronaviruses, for instance in the 5' and 3' untranslated regions (UTRs). Viral UTRs are known to play an important role in both pro- and anti-viral responses and recent evidence suggests that evolution of the 3′ UTR is contributing to increased viral diversity (74). Indeed, the 3′ UTR of SARS-CoV-2 shows a severe truncation when compared to SARS-CoV-1 and MERS. Given that viral UTRs are not under selective pressure with respect to a translated protein, they might be more prone to acquire mutations that modulate regulation through host RBPs. Figure 5B and 5C shows predicted RBP binding sites to the 3′ and 5′ UTRs across selected coronaviruses, respectively. While SARS-CoV-1, SARS-CoV-2 and MERS show conserved predicted binding on the 5′ UTR and cluster closely, a depletion of predicted RBP binding sites is observed in the 3′ UTR of SARS-CoV-2 when compared to SARS-CoV-1 and MERS. To investigate gain- and loss-of-binding in viral UTRs across the severe pathogenic human coronaviruses SARS-CoV-1, SARS-CoV-2 and MERS, we performed multiple sequence alignment of the viral 3' and 5' UTRs and compared the predicted binding score profiles across the three viruses.

Loss of FXR2-binding in SARS-CoV-2 3′ UTR. Figure 5E shows predicted 3′ UTR binding of FXR2, a paralog of FMRP (fragile X mental retardation protein). Our model predicted extensive binding of FXR2 along the 3′ UTR of SARS-CoV-1 and MERS, while SARS-CoV-2 showed a complete lack of predicted FXR2 binding sites, owing to its significantly shorter 3′ UTR. On the other hand, Figure 5G shows that predicted FXR2 binding is conserved in the 5' UTR of SARS-CoV-1 and SARS-CoV-2. FXR2 paralog FMRP was previously shown to broadly bind along the entirety the 3′ UTR of the Zika virus (ZIKV) (75). However, while FMRP was suggested to act as a ZIKV restriction factor by blocking viral RNA translation, a significantly reduced ZIKV infection was observed upon knockdown of FXR2 (75).

Predicted FTO binding site is conserved in the 3′ UTR of SARS-CoV-1 and SARS-CoV-2. Altered expression levels of methyltransferase-like 3 (METTL3) and fat mass and obesity-associated protein (FTO) have been recently linked to viral replication (76). FTO is a demethylase (eraser) enzyme with enriched binding in the 3′ UTR of mRNAs in mammals (77). FTO has previously been suggested as a potential drug target against COVID-19 (78), as targeted knockdown has been shown to significantly decrease SARS-CoV-2 infection (76,78,79). Therefore, we investigated predicted binding of FTO to the 3′ UTR of SARS-CoV-2 and related viruses. Indeed, we observed that SARS-CoV-1, SARS-CoV-2 and MERS, as well as the less pathogenic viruses HCoV-HKU1 and HCoV-OC43 harbor at least one predicted FTO binding site in their 3′ UTR (Figure 5B). Further, Figure 5D shows that while SARS-CoV-1 and MERS harbor multiple shared predicted FTO binding sites along their 5′ UTR, SARS-CoV-2 only harbors one predicted FTO binding site at the 3′ end of its 5′ UTR which is exclusively shared with SARS-CoV-1.

Predicted TARDBP binding is newly acquired in the SARS-Cov-2 5′ UTR. We next focus on TARDBP (also known as TDP-43) (Figure 5F), which was predicted to bind the 5′ UTR of a SARS-CoV-2 mutant in a recent study (64). TARDBP, a host protein implicated in pre-mRNA alternative splicing, has been shown to play a role in viral replication and pathogenesis in the context of coxsackievirus B3 infection (80). In contrast to the findings of Mukherjee *et al.* (64), our model predicted a TARDBP binding site at the genomic range of 89-98 in the wild-type reference of SARS-CoV-2. Interestingly, in addition to observing a lack of predicted binding signal of TARDBP on the 5′ UTR of SARS-CoV-1 and MERS, we found a complete lack of predicted TARDBP binding to the 5' UTR of any of the other investigated coronaviruses (Figure 5C). This suggests that 5′ UTR TARDBP binding potential may be newly acquired in SARS-CoV-2 and may affect its virulence.

A functional catalog of human RBPs with predicted SARS-CoV-2 interaction

To understand the potential functional impact of predicted RBP binding sites on the SARS-CoV-2-mediated COVID-19 disease, we set out to interrogate the breadth of pub-

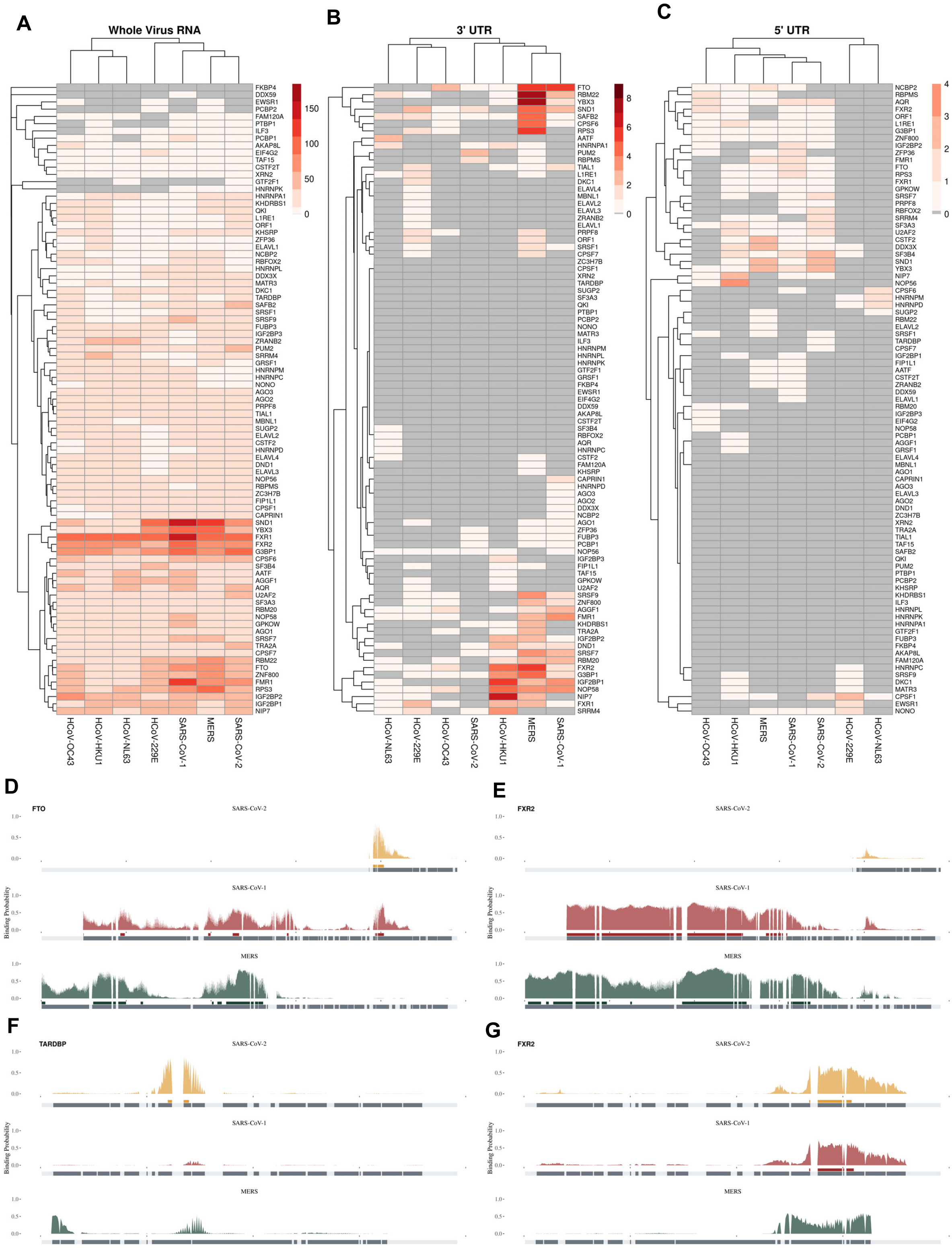


Figure 5. Comparison of SARS-CoV-2 and 6 other human coronaviruses. (A−C) Binding sites were predicted over the seven human coronaviruses, and their number counted over the entire genome (**A**) or over the 3' (**B**) and 5' (**C**) UTRs. Hierarchical clustering was applied to evaluate the proximity between viruses in terms of predicted binding sites composition. (D−G) Examples of evolutionary conserved, gained, and lost binding sites between the three high-severity viruses MERS, SARS-CoV-1, and SARS-CoV2. Panel **D** shows an example for predicted FTO binding sites found only in SARS-Cov-2 and SARS-CoV-1 in their 3' UTRs. Panel **E** shows a predicted binding site for FXR2 only shared between MERS and SARS-CoV-1 in their 3' UTR. Panel **F** shows a predicted binding site for TARDBP exclusive to SARS-CoV-2 in the 5' UTR. Panel **G** shows a predicted binding site for FXR2 only shared between SARS-CoV-2 and SARS-CoV-1 in the 5' UTR.

licly available OMIC research, thereby gathering supportive evidences for our 88 RBPs models (Figure 6). We collected 97 data sets across 22 studies, covering experimentally determined and predicted viral RNA - host RBP interactions as well as multi-level (OMICS) data related to SARS-CoV-2 cell line infections, shedding light on viral entry, protein–protein interactions and host cell regulation (Methods). Studies which are closer to disease phenotypes, like CRISPR cell survival assays and COVID-19 patient data, were also included. In addition, we collected evidence of direct involvement of RBPs in SARS-CoV-2 infection, as reported in the SIGNOR database, a manually curated resource of pathways and genes involved in SARS-CoV-2 (81). Through data integration we, report evidence of binding or regulation for 85 out of 88 RBP models.

We found that a large fraction (64 out of 88, 72.4%) of RBPs were identified to directly bind SARS-CoV-2 RNA using affinity-purification methods (18,19) (labeled as *Experimentally supported binders*, Figure 6), supporting the predicted interactions of these RBPs with the viral RNA. Only 32 out of our 88 RBPs (36.8%) were common to RBPs reported by other predictive methods, namely catRAPID (49) or PRISMNet (50). However, these methods predict and report only RBPs binding the viral UTRs and the spike S gene. In addition, they do not provide single nucleotide resolution of the predicted binding sites, therefore the number and location of our predicted sites is not directly comparable to those studies. We thus complement the knowledge on predicted binding site locations over SARS-CoV-2 RNA with 56 RBPs uniquely reported by our framework, 37 of which are experimentally supported for viral RNA interactions. Our holistic comparison revealed that the majority of explored RBPs were previously reported to be part of host-pathogen PPI networks, cellular pathways which are altered during infection by either SARS-CoV-2, SARS-CoV-1 or CRISPR knock-out screenings, including 17 RBPs with no experimental evidence of direct binding to SARS-CoV-2 (labeled as *Infection relevant predicted binders*, Figure 6). This highlights the importance of RBPs in the infection process, immune response and viral replication. Although no RBP co-localized with loci associated to COVID-19 severe disease courses (GWAS) under genome-wide significance, we identified 44 (50.6%) RBPs with nominal significance. When considering the total of 2,730 coding genes co-localizing nominally associated loci, this represents a significant enrichment for RBPs (odds ratio of 7.8, Fisher test *P*-value $< 2.2e{-}16$), suggesting their importance in patient's course. Finally, a small set of our predicted binding RBPs was shown to be supported only from CRISPR screens or found deregulated across COVID-19 patients, without experimental evidence in molecular networks (labeled as *Disease relevant predicted binders*, Figure 6). Taken together, the large overlap between our predicted RBP binders and the different resources considered confirms that hijacking host RBPs might be crucial to the infection life cycle of the virus.

DISCUSSION

Strong evidence suggests that human RBPs are critical host factors for viral infection by SARS-CoV-2, yet experimental data on the exact binding sites of RBPs across the SARS-CoV-2 genome is still parse, due to the high costs associated with performing large-scale CLIP-seq experiments. To combat this knowledge gap, we constructed the first *in silico* human-virus RBP-RNA interaction map for SARS-CoV-2 using predictions from pysster (25) and DeepRiPe (26) models trained on a large cohort of eCLIP and PAR-CLIP datasets, respectively. The use of high-capacity CNN classifiers represents a significant improvement over previous computational studies performing motif scanning over the SARS-CoV-2 genome (40,82), as it enables the learning of more complex binding syntax and thus the successful prediction of binding sites for RBPs with no clear sequence motif. This is evident by the fact that we observed high performance for RBPs without annotations of binding motifs in literature. On the other hand, we recovered known binding motifs for several RBPs (including QKI, RBFOX2 and TARDBP) using gradient-based attribution methods. Together with stringent performance evaluation and conservative selection of high-quality models, these results suggest that our models predict binding sites with high accuracy on the basis of genuine RNA sequence sequence. In a recent study, the PRISMNet deep learning model was used to infer binding of 42 host RBPs to the SARS-CoV-2 genome (50). However, PRISMNet predictions were restricted to the 5' and 3' viral UTR regions and are rather large, with some spanning over hundreds of nucleotides, while RBP binding usually only occurs across short stretches of RNA *in vivo*. In contrast, our approach predicted single-nucleotide binding probabilities across the entire viral genome and may therefore yield a more accurate picture of the binding landscape of human RBPs to SARS-CoV-2.

Our study predicted binding sites for RBPs known to interact with SARS-CoV-2, as well as RBPs without existing experimental evidence to directly binding to the SARS-CoV-2 RNA. Further, the predicted binding map provides a rich resource for future functional studies, in particular for investigating the role of the SARS-CoV-2 protein–RNA interactome in context of the viral life cycle. For instance, binding site predictions may be used to accelerate the discovery of host RBPs that engage in both pro- and anti-viral functions by directly interacting with the viral RNA. Further, predictions may aid in the identification of functional sites on the viral RNA that can be therapeutically targeted by RNA drugs, such as anti-sense oligonucleotides, to interfere with host RBP binding. In addition to constructing a RBP binding map based on predicted binding sites on the SARS-CoV-2 reference sequence, we quantified the impact of sequence variants via *in silico* mutagenesis across 11 SARS-CoV-2 strains, including the alpha, delta and omicron viral strains.

Additionally, we performed systematic *in silico* mutagenesis of all positions in the SARS-CoV-2 genome, pinpointing mutations associated with particularly high impact, which could represent potential high-risk variants to monitor in the future. Through computation of feature importance scores on the reference and alternative sequence, our method revealed how sequence variants impact protein–RNA interactions. In previous studies, variants of concerns have been prioritized through their potential impact on the sequence of viral proteins, in particular the Spike protein.

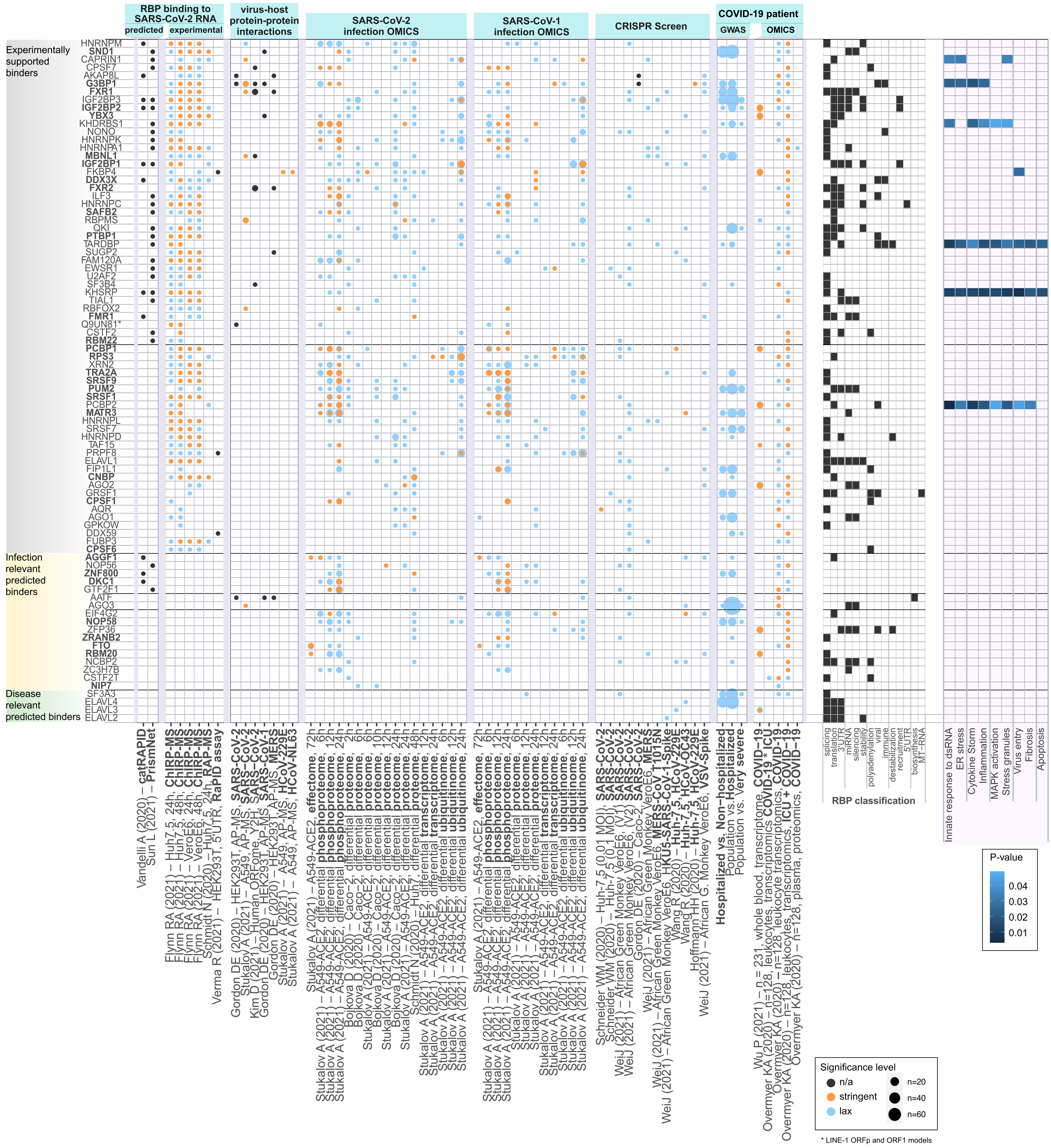


Figure 6. RBPs in context of public *in vitro* and patient OMICS data. RBP with model predictions (rows) annotated with experimental evidences found in 92 multiOMIC publicly available research results (columns) followed by information from RBP classification and role in known SARS-CoV-2 pathways. From left to right: RBPs were manually assigned to three categories according to their annotation pattern. RBPs predicted to bind SARS-CoV-2 RNA by the other prediction methods catRAPID, PrismNET. RBPs binding to SARS-CoV-2 RNA determined experimentally by ChIRP-MS, RAP-MS and RaPID assay. Evidences of RBPs with stringent or lax significance cutoffs found in further 55 data sets across multiple OMICS levels and experiment types were grouped by experimental context: Experimental viral-host protein interactions measured by AP-MS across various coronaviruses, SARS-CoV-2 and SARS-CoV-2 infection OMIC (timecourses), selected CRISPR studies, most recent GWAS data (release 6) by Host Genetics Initiative and blood-based patient OMICS data. Classification of RBP according to their roles related to biological processes. Far right: Annotation of RBPs to pathways related to SARS-CoV-2 infections obtained from SIGNOR database.

Our results complement these finding by giving insight into how strain-defining sequence mutations of variants of concern affect RNA regulatory elements, which may explain their impact on viral efficiency. With our comparative analysis of predicted RBP-RNA interactions across seven coronaviruses, we contribute towards the identification of RNA regulatory elements that are exclusive to SARS-CoV-2 and may therefore modulate its transmission and pathogenicity, compared to SARS-CoV-1, MERS and less pathogenic coronaviruses. Both the variants of concern and comparative analysis highlight predicted gain- or loss-of-binding events and therefore pinpoint RBPs which can be prioritized for further screening.

We integrated knowledge of RBPs with predicted binding sites on the SARS-CoV-2 RNA across other pathogens, host-viral protein–protein interactions, numerous studies collecting functional and phenotypic data, such as GWAS and CRISPR screens, as well as multi-omics COVID-19 patient data, in order to pinpoint RBPs with potential clinical significance.

Among RBPs with predicted binding sites on SARS-CoV-2 which are part of experimental protein–RNA interaction assays, we find several known regulators of viral processes. For example, we find the viral restriction factor hnRNPR (66), the IGF2BP1-3 RBPs, which are linked, through GWAS, to poor disease outcome (83), key regulators of SARS-CoV-2 infection CAPRIN1 and KHDRBS1 (84), the pro-viral factor pro-DDX3X (61) and the host factor NONO (66). Predicted binding of many RBPs was impacted by mutations in SARS-CoV-2 variants of concern, or showed changes when compared to other coronaviruses. For instance, TARDBP may be of particularly interest in the context of SARS-CoV-2 infection due to a unique predicted binding site in the virus 5' UTR, compared to SARS-CoV-1, MERS and other coronaviruses.

Another RBP of interest is the Serine/arginine-rich splicing factor 7 (SRSF7), previously shown to interact with coronavirus RNA (85). It has been suggested that this spliceosome protein could be sequestrated by the viral genome, the later thus acting as a sponge through these putative binding sites, to alter host splicing processes. Among the high-impact mutations in the SRSF7 gene, position 23,604 (S protein gene) is found mutated across multiple strains, with different alternative nucleotides: a C>A transversion is found in alpha and mu variants, while a C>G transversion is found in delta and kappa variants. Both mutations are associated to a positive delta score, i.e. a predicted gain-of-binding. Besides SRSF7, the large number of predicted binding sites for splicing factors at the 5' UTR of the SARS-CoV-2 (cluster 6, Figure 3C) and the significant enrichment of host and viral restriction RBPs in cluster 4 (Figure 3C), might suggest that these RBPs are likely to get sponged on the viral genome and by that modulate post-transcriptional regulatory networks in the host cell.

One other interesting RBP is represented by FXR2, paralog of FXR1 and FMR1 which are experimentally identified as direct binders of SARS-CoV-2 (Figure 6). Recent evidence suggests that FXR2 selectively interact with MERS viral proteins but not with viral proteins from SARS-CoV-1 and SARS-CoV-2 (28). While we find evidence of predicted FXR2 binding along the SARS-CoV-2 genome, this is in agreement with the results of our comparative analysis with other human coronaviruses, where we predict extensive binding of FXR2 along the 3′ UTR of SARS-CoV-1 and MERS, but depletion of FXR2 binding in the SARS-CoV-2 3′ UTR. Together with the evidence of genetic association of FXR2 to COVID-19 disease severity (86), our findings might suggest a role of FXR2 regarding the severity of the infection, although the physiological relevance of this RBPs remains to be experimentally assessed.

SARS-Cov-2 utilizes the endoplasmic reticulum (ER)-derived double membrane vesicles (DMVs) as replication centers. RNA viruses, included SARS-CoV-2, contains several instances of an RNA regulatory motif, called SECReTE motif (67) which facilitates localization to the ER and increases viral protein translation, as well as viral replication. Such motif is also found in some human mRNAs encoding for proteins involved in innate immunity and associated with epithelial layers targeted by SARS-CoV-2. This suggests that host and pathogen might compete for ER-associated RBPs. Indeed, we identified two clusters (7 and 8, Figure 3C) which harbor RBPs with binding sites significantly overlapping with SECReTE motifs. These include FUBP3, KHSRP and MATR3 (cluster 8), already identified previously as important host or restriction factors for other RNA virus infections (66). Interestingly, we linked MATR3 to several CRISPR studies showing that this factor is essential for SASR-CoV-2 replication, as well as to many nominal variants in all GWAS data (Figure 6). MATR3 physically interacts with G3BP1, another predicted RBP in this set which has been found to interact specifically with the SARS-CoV-2 nucleocapsid (N) protein, controls viral replication and localizes (together with MATR3) at stress granules where G3BP3 is taken away from its typical interactions partners, thereby impairing stress granule formation (87). The fact that G3BP3 binding is enriched in correspondence of the gene encoding for protein N (Figure 3C) might also suggest a direct regulation of this transcript by G3BP3 in a feedback loop manner.

The fat mass and obesity-associated protein (FTO) is an example of a RBP with predicted binding sites that has not been previously suggested to bind to the SARS-CoV-2 RNA. Besides the predicted binding pattern, FTO also presented numerous important gain- or loss-of-binding across many viral strains. Although there was no clear trend towards systematic loss-of-binding of FTO across the viral variants, we were able to point out multiple close-by mutations in the alpha variant that were predicted to be associated to a significant loss, around the position 28,280 (Figure 4B). Finally, the FTO protein was identified as key risk factor for obesity by other studies (Figure 6, GWAS, variant lowest P-value 0.0053), which is also a known risk for COVID-19 severity. A small set of RBPs showed predicted binding sites by our approach, while showing little to no experimental evidence across multiple functional studies. This includes the ELAVL2-4 factors (Figure 6). Interestingly, ELAVL2-4 RBPs, predicted from our analysis to be SECReTE motif-associated RBPs, were also found to be deregulated in COVID-19 patients (Figure 6). These RBPs might represent promising candidates whose molecular mechanisms can be further investigated experimentally.

Our study provides a resource of computationally predicted binding sites of human RBPs to the SARS-CoV-2 RNA at high resolution - information that is currently not available through MS-based pull-down experiments. It is, however, important to point out some of the limitations of our approach. First, our method predicts the set of *potential* sites that may be bound by a protein of interest, rather than a specific configuration of sites occupied with binding proteins *in vivo*, with the later being, among other factors, highly cell-type specific. Further experiments are required to assess the role of those proteins (and their predicted binding sites) in a SARS-CoV-2 infection context. Second, as demonstrated in the comparison with CNBP eCLIP experimental data, our method misses some of the experimentally observed binding sites (Supplementary Figure S1). We believe that a certain fraction of false negative predictions correspond to viral-specific structures recognized by a host RBP, with a specific sequence composition which was probably never observed in human peaks and therefore not captured by a model trained on human data. It remains to be demonstrated whether future improvements to our current method, to incorporate knowledge of viral-specific sequences bound by host RBPs in the learning framework, could improve the method's recall. All in all, our resource will help to prioritize RBPs for experimental investigation and direct future efforts towards promising candidates.

CONCLUSION

Viruses depend on essential host factors at all stages of their infection cycle. One family of host factors, RNA-binding proteins (RBPs), are involved in multiple aspects of post-transcriptional regulation. While several RBPs have been associated with SARS-CoV-2, some of which may represent drug-able targets for anti-viral therapy, cost and time constraints render a comprehensive experimental profiling of human RBPs to the SARS-CoV-2 RNA at high spatial resolution infeasible. Mapping binding sites of human RBPs to the exact locations in the viral RNA is, however, crucial for gaining a mechanistic understanding of viral interaction with the host's post-transcriptional machinery. We used the pysster and DeepRiPe deep learning frameworks together with data from CLIP-seq experiments to create an *in silico* binding map of human RBPs along the SARS-CoV-2 genome at single-nucleotide resolution. Predicted binding profiles of RBPs suggested that groups of RBPs exhibit similar binding patterns on the viral genome and that RBPs within these group may be functionally related, for example, by being associated to the SECReTE motif - a motif previously associated with efficient viral replication. We further utilized trained models to score the impact of strain-defining sequence variants across 11 SARS-CoV-2 strains and predicted several gain-or loss-of-binding events, some of which simultaneously impact the binding of multiple RBPs or are conserved in multiple viral strain. In addition, we quantified the impact of hypothetical variants by performing extensive *in silico* mutagenesis, thereby generating all possible point mutations across the SARS-CoV-2 genome. Finally, we predicted RBP-binding across 6 other human coronaviruses (including SARS-CoV-1 and MERS) and identified several conserved binding site predictions as well newly predicted binding sites in SARS-CoV-2. All generated results, including fully trained models, predicted binding sites across SARS-CoV-2 and other coronaviruses and variant impact scores are publicly available at ⟨0:italic ⟩https://sc2rbpmap.helmholtz-muenchen.de/⟨/0:italic⟩. We believe that our results can help to give new insights into the role of RNA-binding proteins in context of SARS-CoV-2 infection and represent a rich resource for further research on how SARS-CoV-2 hijacks the host cell's RNA regulatory machinery.

DATA AVAILABILITY

Pre-trained models, together with scripts for training and prediction as well as binding site predictions and variant impact scores are available at https://github.com/mhorlacher/sc2rbpmap and https://doi.org/10.5281/zenodo.7547354. Graphical representations of predicted RBP binding sites on the SARS-CoV-2 genome and variant impact scores for variants of concern are available at https://sc2rbpmap.helmholtz-munich.de.

SUPPLEMENTARY DATA

Supplementary Data are available at NARGAB Online.

FUNDING

Helmholtz Association under the joint research school 'Munich School for Data Science (MUDS)' (to M.H., S.O., G.C., P.S., A.M.); Deutsche Forschungsgemeinschaft [SFB/TR501 84 TP C01 to A.M., L.M.]; Helmholtz Association aeroHEALTH (to A.M., Y.H.); Joachim Herz Foundation (to Y.H., M.G.); U.O. are supported by the Berlin Center of Machine Learning (BZML) funded by the German Ministry for Education and Research (BMBF).

Conflict of interest statement. F.B. and N.S.M. hold positions at knowing01 GmbH that might benefit or be at a disadvantage from the published findings. The remaining authors declare no conflict of interest that is relevant to the content of this article.

REFERENCES

1. V'kovski,P., Kratzel,A., Steiner,S., Stalder,H. and Thiel,V. (2021) Coronavirus biology and replication: implications for SARS-CoV-2. *Nat. Rev. Microbiol.*, **19**, 155–170.
2. Sola,I., Almazán,F., Zúñiga,S. and Enjuanes,L. (2015) Continuous and discontinuous RNA synthesis in coronaviruses. *Annu. Rev. Virol.*, **1**, 265–88.
3. Kim,D., Lee,J., Yang,J., Kim,J., Kim,V. and Chang,H. (2020) The architecture of SARS-CoV-2 transcriptome. *Cell*, **181**, 914–921.
4. Davey,N.E., Travé,G. and Gibson,T.J. (2011) How viruses hijack cell regulation. *Trends biochem. sci.*, **36**, 159–169.
5. Fung,T.S. and Liu,D.X. (2019) Human coronavirus: host-pathogen interaction. *Ann. Rev. Microbiol.*, **73**, 529–557.
6. Xiang,J.S., Mueller,J.R., Luo,E.-C., Yee,B., Schafer,D., Schmok,J.C., Tan,F.E., Her,H.-L., Chen,C.-Y., Brannan,K.W. *et al.* (2021) Discovery and functional interrogation of the virus and host RNA interactome of SARS-Cov-2 proteins. SSRN Scholarly Paper 3867726, Social Science Research Network Rochester, NY.
7. Stukalov,A., Girault,V., Grass,V., Karayel,O., Bergant,V., Urban,C., Haas,D.A., Huang,Y., Oubraham,L., Wang,A. *et al.* (2021) Multilevel proteomics reveals host perturbations by SARS-CoV-2 and SARS-CoV. *Nature*, **594**, 246–252.

8. Gordon,D.E., Jang,G.M., Bouhaddou,M., Xu,J., Obernier,K., White,K.M., O'Meara,M.J., Rezelj,V.V., Guo,J.Z., Swaney,D.L. *et al.* (2020) A SARS-CoV-2 protein interaction map reveals targets for drug repurposing. *Nature*, **583**, 459–468.
9. Wei,J., Alfajaro,M.M., DeWeirdt,P.C., Hanna,R.E., Lu-Culligan,W.J., Cai,W.L., Strine,M.S., Zhang,S.-M., Graziano,V.R., Schmitz,C.O. and et,al. (2021) Genome-wide CRISPR screens reveal host factors critical for SARS-CoV-2 Infection. *Cell*, **184**, 76–91.
10. Hentze,M.W., Castello,A., Schwarzl,T. and Preiss,T. (2018) A brave new world of RNA-binding proteins. *Nat. Rev. Mol. Cell Biol.*, **19**, 327–341.
11. Li,Z. and Nagy,P.D. (2011) Diverse roles of host RNA binding proteins in RNA virus replication. *RNA Biol*, **8**, 305–315.
12. Molleston,J.M. and Cherry,S. (2017) Attacked from all sides: RNA decay in antiviral defense. *Viruses*, **9**, 2.
13. Garcia-Moreno,M., Järvelin,A.I. and Castello,A. (2018) Unconventional RNA-binding proteins step into the virus–host battlefront. *Wiley Interdiscipl. Rev.: RNA*, **9**, e1498.
14. Manokaran,G., Finol,E., Wang,C., Gunaratne,J., Bahl,J., Ong,E., Tan,H., Sessions,O., Ward,A., Gubler,D. *et al.* (2015) Dengue subgenomic RNA binds TRIM25 to inhibit interferon expression for epidemiological fitness. *Science*, **9**, 6257.
15. Vashist,S., Urena,L., Chaudhry,Y. and Goodfellow,I. (2012) Identification of RNA-protein interaction networks involved in the norovirus life cycle. *J. Virol.*, **86**, 11977–11990.
16. Luo,H., Chen,Q., Chen,J., Chen,K., Shen,X. and Jiang,H. (2005) The nucleocapsid protein of SARS coronavirus has a high binding affinity to the human cellular heterogeneous nuclear ribonucleoprotein A1. *FEBS Lett.*, **579**, 2623–2628.
17. Wu,C.-H., Chen,P.-J. and Yeh,S.-H. (2014) Nucleocapsid phosphorylation and RNA helicase DDX1 recruitment enables coronavirus transition from discontinuous to continuous transcription. *Cell Host Microbe*, **16**, 462–472.
18. Flynn,R.A., Belk,J.A., Qi,Y., Yasumoto,Y., Wei,J., Alfajaro,M.M., Shi,Q., Mumbach,M.R., Limaye,A., DeWeirdt,P.C. *et al.* (2021) Discovery and functional interrogation of SARS-CoV-2 RNA-host protein interactions. *Cell*, **184**, 2394–2411.
19. Schmidt,N., Lareau,C.A., Keshishian,H., Ganskih,S., Schneider,C., Hennig,T., Melanson,R., Werner,S., Wei,Y., Zimmer,M. *et al.* (2021) The SARS-CoV-2 RNA–protein interactome in infected human cells. *Nat. Microbiol.*, **6**, 339–353.
20. Lee,S., Lee,Y.-S., Choi,Y., Son,A., Park,Y., Lee,K.-M., Kim,J., Kim,J.-S. and Kim,V.N. (2021) The SARS-CoV-2 RNA interactome. *Mol. Cell.*, **81**, 2838–2850.
21. Labeau,A., Fery-Simonian,L., Lefevre-Utile,A., Pourcelot,M., Bonnet-Madin,L., Soumelis,V., Lotteau,V., Vidalain,P.-O., Amara,A. and Meertens,L. (2022) Characterization and functional interrogation of SARS-CoV-2 RNA interactome. *Cell Rep.*, **39**, 110744.
22. Hafner,M., Katsantoni,M., Köster,T., Marks,J., Mukherjee,J., Staiger,D., Ule,J. and Zavolan,M. (2021) CLIP and complementary methods. *Nat. Rev. Meth. Prim.*, **1**, 20.
23. Van Nostrand,E., Freese,P., Pratt,G., Wang,X., Wei,X., Xiao,R., Blue,S., Chen,J., Cody,N. and Dominguez,D. (2020) A large-scale binding and functional map of human RNA-binding proteins. *Nature*, **583**, 711–719.
24. Van Nostrand,E.L., Freese,P., Pratt,G.A., Wang,X., Wei,X., Xiao,R., Blue,S.M., Chen,J.-Y., Cody,N.A., Dominguez,D. *et al.* (2020) A large-scale binding and functional map of human RNA-binding proteins. *Nature*, **583**, 711–719.
25. Budach,S. and Marsico,A. (2018) Pysster: classification of biological sequences by learning sequence and structure motifs with convolutional neural networks. *Bioinformatics*, **34**, 3035–3037.
26. Ghanbari,M. and Ohler,U. (2020) Deep neural networks for interpreting RNA-binding protein target preferences. *Genome Res.*, **30**, 214–226.
27. Pairo-Castineira,E., Clohisey,S., Klaric,L., Bretherick,A.D., Rawlik,K., Pasko,D., Walker,S., Parkinson,N., Fourman,M.H., Russell,C.D. *et al.* (2021) Genetic mechanisms of critical illness in COVID-19. *Nature*, **591**, 92–98.
28. Gordon,D.E., Hiatt,J., Bouhaddou,M., Rezelj,V.V., Ulferts,S. *et al.* (2020) Comparative host-coronavirus protein interaction networks reveal pan-viral disease mechanisms. *Science*, **370**, eabe9403.
29. Verma,R., Saha,S., Kumar,S., Mani,S., Maiti,T.K. and Surjit,M. (2021) RNA-protein interaction analysis of SARS-CoV-2 5' and 3' untranslated regions reveals a role of lysosome-associated membrane protein-2a during viral infection. *mSystems*, **6**, e0064321.
30. D'Alessandro,A., Thomas,T., Dzieciatkowska,M., Hill,R.C., Francis,R.O., Hudson,K.E., Zimring,J.C., Hod,E.A., Spitalnik,S.L. and Hansen,K.C. (2020) Serum proteomics in COVID-19 patients: Altered coagulation and complement status as a function of IL-6 level. *J. Proteome Res.*, **19**, 4417–4427.
31. Demichev,V., Tober-Lau,P., Lemke,O., Nazarenko,T., Thibeault,C., Whitwell,H., Röhl,A., Freiwald,A., Szyrwiel,L., Ludwig,D. *et al.* (2021) A time-resolved proteomic and prognostic map of COVID-19. *Cell Syst.*, **12**, 780–794.
32. Di,B., Jia,H., Luo,O.J., Lin,F., Li,K., Zhang,Y., Wang,H., Liang,H., Fan,J. and Yang,Z. (2020) Identification and validation of predictive factors for progression to severe COVID-19 pneumonia by proteomics. *Signal Transduct. Target. Ther.*, **5**, 217.
33. Geyer,P.E., Arend,F.M., Doll,S., Louiset,M.-L., Virreira Winter,S., Müller-Reif,J.B., Torun,F.M., Weigand,M., Eichhorn,P., Bruegel,M. *et al.* (2021) High-resolution serum proteome trajectories in COVID-19 reveal patient-specific seroconversion. *EMBO Mol. Med.*, **13**, e14167.
34. Messner,C.B., Demichev,V., Wendisch,D., Michalick,L., White,M., Freiwald,A., Textoris-Taube,K., Vernardis,S.I., Egger,A.-S., Kreidl,M. *et al.* (2020) Ultra-high-throughput clinical proteomics reveals classifiers of COVID-19 infection. *Cell Syst.*, **11**, 11–24.
35. Overmyer,K., Shishkova,E., Miller,I., Balnis,J., Bernstein,M., Peters-Clarke,T., Meyer,J., Quan,Q., Muehlbauer,L., Trujillo,E. *et al.* (2021) Large-scale multi-omic analysis of COVID-19 severity. *Cell Syst*, **12**, 23–40.
36. Shen,B., Yi,X., Sun,Y., Bi,X., Du,J., Zhang,C., Quan,S., Zhang,F., Sun,R., Qian,L. *et al.* (2020) Proteomic and metabolomic characterization of COVID-19 patient Sera. *Cell*, **182**, 59–72.
37. Wu,P., Chen,D., Ding,W., Wu,P., Hou,H., Bai,Y., Zhou,Y., Li,K., Xiang,S., Liu,P. *et al.* (2021) The trans-omics landscape of COVID-19. *Nat. Commun.*, **12**, 4543.
38. Sugimoto,Y., König,J., Hussain,S., Zupan,B., Curk,T., Frye,M. and Ule,J. (2012) Analysis of CLIP and iCLIP methods for nucleotide-resolution studies of protein–RNA interactions. *Genome Biol.*, **13**, R67.
39. Wheeler,E.C., Van Nostrand,E.L. and Yeo,G.W. (2018) Advances and challenges in the detection of transcriptome-wide protein–RNA interactions. *Wiley Interdiscipl. Rev.: RNA*, **9**, e1436.
40. Srivastava,N., Hinton,G., Krizhevsky,A., Sutskever,I. and Salakhutdinov,R. (2014) Dropout: a simple way to prevent neural networks from overfitting. *J. Mach. Lear. Res.*, **15**, 1929–1958.
41. Kingma,D.P. and Ba,J. (2014) Adam: a method for stochastic optimization. arXiv doi: https://arxiv.org/abs/1412.6980, 22 December 2014, preprint: not peer reviewed.
42. Dominguez,D., Freese,P., Alexis,M.S., Su,A., Hochman,M., Palden,T., Bazile,C., Lambert,N.J., Van Nostrand,E.L., Pratt,G.A. *et al.* (2018) Sequence, structure, and context preferences of human RNA binding proteins. *Mol. Cell*, **70**, 854–867.
43. Sundararajan,M., Taly,A. and Yan,Q. (2017) Axiomatic attribution for deep networks. In: *International Conference on Machine Learning*. PMLR, pp. 3319–3328.
44. Fernandes,J.D., Hinrichs,A.S., Clawson,H., Gonzalez,J.N., Lee,B.T., Nassar,L.R., Raney,B.J., Rosenbloom,K.R., Nerli,S., Rao,A.A. *et al.* (2020) The UCSC SARS-CoV-2 Genome Browser. *Nat. Genet.*, **52**, 991–998.
45. Sayers,E.W., Beck,J., Bolton,E.E., Bourexis,D., Brister,J.R., Canese,K., Comeau,D.C., Funk,K., Kim,S., Klimke,W. *et al.* (2021) Database resources of the national center for biotechnology information. *Nucleic Acids Res.*, **49**, D10.
46. Sievers,F. and Higgins,D.G. (2014) Clustal Omega, accurate alignment of very large numbers of sequences. In: *Multiple Sequence Alignment Methods*. Springer, pp. 105–116.
47. Stelzer,G., Rosen,N., Plaschkes,I., Zimmerman,S., Twik,M., Fishilevich,S., Stein,T.I., Nudel,R., Lieder,I., Mazor,Y. *et al.* (2016) The GeneCards suite: from gene data mining to disease genome sequence analyses. *Curr. Prot. Bioinform.*, **54**, 1.30.1–1.30.33.
48. knowing01, https://knowing01.com/.
49. Vandelli,A., Monti,M., Milanetti,E., Armaos,A., Rupert,J., Zacco,E., Bechara,E., Delli Ponti,R. and Tartaglia,G.G. (2020) Structural

analysis of SARS-CoV-2 genome and predictions of the human interactome. *Nucleic Acids Res.*, **48**, 11270–11283.
50. Sun,L., Li,P., Ju,X., Rao,J., Huang,W., Ren,L., Zhang,S., Xiong,T., Xu,K., Zhou,X. *et al.* (2021) In vivo structural characterization of the SARS-CoV-2 RNA genome identifies host proteins vulnerable to repurposed drugs. *Cell*, **184**, 1865–1883.
51. Kim,D.-K., Weller,B., Lin,C.-W., Sheykhkarimli,D., Knapp,J.J., Kishore,N., Sauer,M., Rayhan,A., Young,V., Marin-de la Rosa,N. *et al.* (2023) A proteome-scale map of the SARS-CoV-2–human contactome. *Nat Biotechnol.*, **14**, 140–149
52. Bojkova,D., Klann,K., Koch,B., Widera,M., Krause,D., Ciesek,S., Cinatl,J. and Münch,C. (2020) Proteomics of SARS-CoV-2-infected host cells reveals therapy targets. *Nature*, **583**, 469–472.
53. Hoffmann,H.-H., Sánchez-Rivera,F.J., Schneider,W.M., Luna,J.M., Soto-Feliciano,Y.M., Ashbrook,A.W., Le Pen,J., Leal,A.A., Ricardo-Lax,I., Michailidis,E. *et al.* (2021) Functional interrogation of a SARS-CoV-2 host protein interactome identifies unique and shared coronavirus host factors. *Cell Host Microbe*, **29**, 267–280.
54. Schneider,W.M., Luna,J.M., Hoffmann,H.-H., Sánchez-Rivera,F.J., Leal,A.A., Ashbrook,A.W., Le Pen,J., Ricardo-Lax,I., Michailidis,E., Peace,A. *et al.* (2021) Genome-scale identification of SARS-CoV-2 and Pan-coronavirus host factor networks. *Cell*, **184**, 120–132.
55. Wang,R., Simoneau,C.R., Kulsuptrakul,J., Bouhaddou,M., Travisano,K.A., Hayashi,J.M., Carlson-Stevermer,J., Zengel,J.R., Richards,C.M., Fozouni,P. *et al.* (2021) Genetic screens identify host factors for SARS-CoV-2 and common cold coronaviruses. *Cell*, **184**, 106–119.
56. Horlacher,M., Wagner,N., Moyon,L., Kuret,K., Goedert,N., Salvatore,M., Ule,J., Gagneur,J., Winther,O. and Marsico,A. (2022) Towards In-Silico CLIP-seq: Predicting Protein-RNA Interaction via Sequence-to-Signal Learning. bioRxiv doi: https://doi.org/10.1101/2022.09.16.508290, 19 September 2022, preprint: not peer reviewed.
57. Jangi,M., Boutz,P., Paul,P. and Sharp,P. (2014) Rbfox2 controls autoregulation in RNA-binding protein networks. *Genes Dev.*, **28**, 637–51.
58. Hallegger,M., Chakrabarti,A., Lee,F., Lee,B., Amalietti,A., Odeh,H., Copley,K., Rubien,J., Portz,B., Kuret,K. *et al.* (2021) TDP-43 condensation properties specify its RNA-binding and regulatory repertoire. *Cell*, **184**, 637–51.
59. Teplova,M., Hafner,M., Teplov,D., Essig,K., Tuschl,T. and Patel,D. (2013) Structure-function studies of STAR family Quaking proteins bound to their in vivo RNA target sites. *Genes Dev.*, **27**, 928–940.
60. Lambert,N., Robertson,A., Jangi,M., McGeary,S., Sharp,P. and Burge,C. (2014) RNA Bind-n-Seq: quantitative assessment of the sequence and structural binding specificity of RNA binding proteins. *Mol. Cell*, **54**, 887–900.
61. Ciccosanti,F., Di Rienzo,M., Romagnoli,A., Colavita,F., Refolo,G., Castilletti,C., Agrati,C., Brai,A., Manetti,F., Botta,L. *et al.* (2021) Proteomic analysis identifies the RNA helicase DDX3X as a host target against SARS-CoV-2 infection. *Antiviral Res.*, **190**, 105064.
62. Mahiet,C. and Swanson,C.M. (2016) Control of HIV-1 gene expression by SR proteins. *Biochem. Soc. Trans.*, **44**, 1417–1425.
63. Tirumuru,N., Zhao,B., Lu,W., Lu,Z., C,H. and Wu,L. (2016) N(6)-methyladenosine of HIV-1 RNA regulates viral infection and HIV-1 Gag protein expression. *Elife*, **5**, e15528.
64. Mukherjee,M. and Goswami,S. (2020) Global cataloguing of variations in untranslated regions of viral genome and prediction of key host RNA binding protein-microRNA interactions modulating genome stability in SARS-CoV-2. *PloS One*, **15**, e0237559.
65. Rothamel,K., Arcos,S., Kim,B., Reasoner,C., Lisy,S., Mukherjee,N. and Ascano,M. (2021) ELAVL1 primarily couples mRNA stability with the 3' UTRs of interferon-stimulated genes. *Cell Rep.*, **35**, 109178.
66. Pennemann,F., Mussabekova,A., Urban,C., Stukalov,A., Andersen,L., Grass,V., Lavacca,T., Holze,C., Oubraham,L., Benamrouche,Y. *et al.* (2021) Cross-species analysis of viral nucleic acid interacting proteins identifies TAOKs as innate immune regulators. *Nat. Commun.*, **12**, 7009.
67. Haimovich,G., Olender,T., Baez,C. and Gerst,J.E. (2020) Identification and enrichment of SECReTE cis-acting RNA elements in the Coronaviridae and other (+) single-strand RNA viruses. bioRxiv doi: https://doi.org/10.1101/2020.04.20.050088, 20 April 2020, preprint: not peer reviewed.
68. Hou,Y.J., Chiba,S., Halfmann,P., Ehre,C., Kuroda,M., Dinnon,K.H. III, Leist,S.R., Schäfer,A., Nakajima,N., Takahashi,K. *et al.* (2020) SARS-CoV-2 D614G variant exhibits efficient replication ex vivo and transmission in vivo. *Science*, **370**, 1464–1468.
69. Ulrich,L., Halwe,N.J., Taddeo,A., Ebert,N., Schön,J., Devisme,C., Trüeb,B.S., Hoffmann,B., Wider,M., Fan,X. *et al.* (2022) Enhanced fitness of SARS-CoV-2 variant of concern Alpha but not Beta. *Nature*, **602**, 307–313.
70. Hu,J., Peng,P., Cao,X., Wu,K., Chen,J., Wang,K., Tang,N. and Huang,A.-L. (2022) Increased immune escape of the new SARS-CoV-2 variant of concern Omicron. *Cell. Mol. Immun.*, **19**, 293–295.
71. Sanyaolu,A., Okorie,C., Marinkovic,A., Haider,N., Abbasi,A.F., Jaferi,U., Prakash,S. and Balendra,V. (2021) The emerging SARS-CoV-2 variants of concern. *Ther. Adv. Inf. Dis.*, **8**, 20499361211024372.
72. Liu,H., Wei,P., Kappler,J.W., Marrack,P. and Zhang,G. (2022) SARS-CoV-2 Variants of Concern and Variants of Interest Receptor Binding Domain Mutations and Virus Infectivity. *Front. Immuno.*, **13**, 825256.
73. Hoffmann,M., Arora,P., Groß,R., Seidel,A., Hörnich,B.F., Hahn,A.S., Krüger,N., Graichen,L., Hofmann-Winkler,H., Kempf,A. *et al.* (2021) SARS-CoV-2 variants B.1.351 and P.1 escape from neutralizing antibodies. *Cell*, **184**, 2384–2393.
74. Farkas,C., Mella,A., Turgeon,M. and Haigh,J.J. (2021) A novel SARS-CoV-2 viral sequence bioinformatic pipeline has found genetic evidence that the viral 3′ untranslated region (UTR) is evolving and generating increased viral diversity. *Front. Microbiol.*, **12**, 665041.
75. Soto-Acosta,R., Xie,X., Shan,C., Baker,C.K., Shi,P.-Y., Rossi,S.L., Garcia-Blanco,M.A. and Bradrick,S. (2018) Fragile X mental retardation protein is a Zika virus restriction factor that is antagonized by subgenomic flaviviral RNA. *Elife*, **7**, e39023.
76. Zhang,X., Hao,H., Ma,L., Zhang,Y., Hu,X., Chen,Z., Liu,D., Yuan,J., Hu,Z. and Guan,W. (2020) Methyltransferase-like 3 modulates severe acute respiratory syndrome coronavirus-2 RNA N6-methyladenosine modification and replication. *mBio*, **12**, e01067-21.
77. Meyer,K.D., Saletore,Y., Zumbo,P., Elemento,O., Mason,C.E. and Jaffrey,S.R. (2012) Comprehensive analysis of mRNA methylation reveals enrichment in 3′ UTRs and near stop codons. *Cell*, **149**, 1635–1646.
78. Zannella,C., Rinaldi,L., Boccia,G., Chianese,A., Sasso,F.C., De Caro,F., Franci,G. and Galdiero,M. (2021) Regulation of m6A Methylation as a New Therapeutic Option against COVID-19. *Pharmaceuticals*, **14**, 1135.
79. Burgess,H.M., Depledge,D.P., Thompson,L., Srinivas,K.P., Grande,R.C., Vink,E.I., Abebe,J.S., Blackaby,W.P., Hendrick,A., Albertella,M.R. *et al.* (2021) Targeting the m6A RNA modification pathway blocks SARS-CoV-2 and HCoV-OC43 replication. *Genes Dev.*, **35**, 1005–1019.
80. Kuo,P.-H., Chiang,C.-H., Wang,Y.-T., Doudeva,L.G. and Yuan,H.S. (2014) The crystal structure of TDP-43 RRM1-DNA complex reveals the specific recognition for UG-and TG-rich nucleic acids. *Nucleic Acids Res.*, **42**, 4712–4722.
81. Licata,L., Lo Surdo,P., Iannuccelli,M., Palma,A., Micarelli,E., Perfetto,L., Peluso,D., Calderone,A., Castagnoli,L. and Cesareni,G. (2020) SIGNOR 2.0, the SIGnaling Network Open Resource 2.0: 2019 update. *Nucleic Acids Res*, **8**, D504–D510.
82. Bartas,M., Brázda,V., Bohálová,N., Cantara,A., Volná,A., Stachurová,T., Malachová,K., Jagelská,E.B., Porubiaková,O., Červeň,J. *et al.* (2020) In-depth bioinformatic analyses of nidovirales including human SARS-CoV-2, SARS-CoV, MERS-CoV viruses suggest important roles of non-canonical nucleic acid structures in their lifecycles. *Front. Microbiol.*, **11**, 1583.
83. Ilias,I., Diamantopoulos,A., Botoula,E., Athanasiou,N., Zacharis,A., Tsipilis,S., Jahaj,E., Vassiliou,A., Vassiliadi,D., Kotanidou,A. *et al.* (2021) Covid-19 and growth hormone/insulin-like growth factor 1: study in critically and non-critically ill patients. *Front. Endocrinol.*, **12**, 644055.
84. Kamel,W., Noerenberg,M., Cerikan,B., Chen,H., Järvelin,A., Kammoun,M., Lee,J., Shuai,N., Garcia-Moreno,M., Andrejeva,A. *et al.* (2021) Global analysis of protein–RNA interactions in

SARS-CoV-2-infected cells reveals key regulators of infection. *Mol. Cell*, **81**, 2851–2867.
85. Srivastava,R., Daulatabad,S.V., Srivastava,M. and Janga,S.C. (2020) Role of SARS-CoV-2 in altering the RNA binding protein and miRNA directed post-transcriptional regulatory networks in humans. *Int. J. Mol. Sci.*, **21**, 7090.
86. COVID-19 Host Genetics Initiative (2022) A first update on mapping the human genetic architecture of COVID-19. *Nature*, **608**, E1–E10.
87. Nabeel-Shah,S., Lee,H., Ahmed,N., Burke,G., Farhangmehr,S., Ashraf,K., Pu,S., Braunschweig,U., Zhong,G., Wei,H. *et al.* (2022) SARS-CoV-2 nucleocapsid protein binds host mRNAs and attenuates stress granules to impair host stress response. *iScience*, **25**, 103562.
88. Mukherjee,N., Wessels,H., Lebedeva,S., Sajek,M., Ghanbari,M., Garzia,A., Munteanu,A., Yusuf,D., Farazi,T., Hoell,J. *et al.* (2019) Deciphering human ribonucleoprotein regulatory networks. *Nucleic Acids Res*, **47**, 570–581.
89. Feng,H., Bao,S., Rahman,M.A., Weyn-Vanhentenryck,S.M., Khan,A., Wong,J., Shah,A., Flynn,E.D., Krainer,A.R. and Zhang,C. (2019) Modeling RNA-binding protein specificity in vivo by precisely registering protein–RNA crosslink sites. *Mol. Cell*, **74**, 1189–1204.

8 BENCHMARKING RBP-BINDING PREDICTION METHODS

A Systematic Benchmark of Machine Learning Methods for Protein-RNA Interaction Prediction

Authors: Marc Horlacher, Giulia Cantini, Julian Hesse, Patrick Schinke, Nicolas Goedert, Shubhankar Londhe, Lambert Moyon, Annalisa Marsico

Journal: Briefings in Bioinformatics, 2023 [2]

Summary: Development of computational methods for protein-RNA interaction prediction is an active area of research, with numerous methods developed over the last decade. Recent methods are predominantly based on deep learning techniques that leverage artificial neural networks to model complex RBP-binding preferences. The task of protein-RNA interaction prediction is commonly formulated as a classification problem, with methods being tasked to predict whether a given RNA sequence is bound or unbound by a protein of interest. Bound regions are usually derived from processing of CLIP data via peak calling. The continuous emergence of new methods and their evaluation on heterogeneous dataset, spanning different CLIP protocols and RBPs, complicates the identification of state-of-the-art approaches. To address these issues, this study reviews 37 RBP-binding prediction methods and subsequently benchmarks 11 representative approaches on data from 3 large CLIP datasets across various protocols, including eCLIP, iCLIP and PAR-CLIP, comprising 313 experiments and 203 unique RBPs. Further, the study proposes a unified pre-processing framework and employs two negative-class sample generation strategies. Specifically, one strategy is agnostic to CLIP biases while the other performs bias-aware sampling, ensuring that methods are evaluated in an unbiased manner. The study demonstrates that deep learning methods outperform shallow learning methods by a significant margin. Furthermore, multi-task methods that jointly predict the binding of multiple RBPs tend to outperform binary classifiers, possibly due to parameter sharing across tasks. While there appears to be no major performance difference across benchmarked deep learning architectures, the choice of model input modalities was found to have a profound effect on predictive performance. For instance, the addition of sequence conservation scores, exon/intron information and larger input sizes greatly increased model performance. Secondary structure information was shown to improve model performance for a small subset of RBPs and was found to be sensitive to the choice of negative samples, with diminished performance in case negatives are sampled from binding sites of other RBPs. Importantly, models which utilized *in vivo* measured structure outperformed those with predicted structure. Lastly, the study highlights that cross-prediction across cell types and CLIP protocols is associated with a significant decline in model performance, suggesting that models may partially learn cell type or protocol-specific information. This effect is shown to be more pronounced for low-performing models. Together, this study establishes an evaluation framework for protein-RNA interaction prediction methods, highlights state-of-the-art approaches, and identifies key design choices to guide and accelerate further method development.

Availability: Data of all utilized CLIP experiments is deposited at `https://github.com/mhorlacher/Benchmark-RBP` in the form of BED files for peak intervals. Further, the repository contains Snakemake [181] workflows to generate the inputs of benchmarked methods. Finally, Conda [182] environments and code is provided to train and evaluate all benchmarked methods.

Contributions: Study design and conceptualization; Gathering of CLIP datasets and data preprocessing; Implementation of the benchmark workflow pipeline with Snakemake [181], including the implementation of negative-class sampling strategies and dataset splits; Model training and evaluation of GraphProt, iDeepS, Pysster, DeepRAM, DeepCLIP and BERT-RBP; Aggregation of results and figure design; Preparation of the manuscript text and formatting of the submission.

Briefings in Bioinformatics, 2023, 1–17
https://doi.org/10.1093/bib/bbad307
Review

A systematic benchmark of machine learning methods for protein–RNA interaction prediction

Marc Horlacher, Giulia Cantini, Julian Hesse, Patrick Schinke, Nicolas Goedert, Shubhankar Londhe, Lambert Moyon and Annalisa Marsico

Corresponding author: Lambert Moyon. E-mail: lambert.moyon@helmholtz-muenchen.de; Annalisa Marsico. E-mail: annalisa.marsico@helmholtz-muenchen.de

Abstract

RNA-binding proteins (RBPs) are central actors of RNA post-transcriptional regulation. Experiments to profile-binding sites of RBPs *in vivo* are limited to transcripts expressed in the experimental cell type, creating the need for computational methods to infer missing binding information. While numerous machine-learning based methods have been developed for this task, their use of heterogeneous training and evaluation datasets across different sets of RBPs and CLIP-seq protocols makes a direct comparison of their performance difficult. Here, we compile a set of 37 machine learning (primarily deep learning) methods for *in vivo* RBP–RNA interaction prediction and systematically benchmark a subset of 11 representative methods across hundreds of CLIP-seq datasets and RBPs. Using homogenized sample pre-processing and two negative-class sample generation strategies, we evaluate methods in terms of predictive performance and assess the impact of neural network architectures and input modalities on model performance. We believe that this study will not only enable researchers to choose the optimal prediction method for their tasks at hand, but also aid method developers in developing novel, high-performing methods by introducing a standardized framework for their evaluation.

Keywords: benchmark, deep learning, RNA-binding proteins, RNA biology.

INTRODUCTION

Out of the over 20 000 annotated human protein-coding genes, at least 1500 are predicted to code for RNA-binding proteins (RBPs) [1]. RBPs are involved in a diverse number of functions, such as export and localization of transcripts, post-transcriptional modification, alternative splicing and translation [2] and play an important role in human diseases, such as cancer, neurodegenerative and metabolic diseases [3]. Uncovering the targets of RBPs is crucial to elucidate their cellular function in health and diseases. Several experimental methods for identifying RBP *in vivo* binding-sites transcriptome-wide have been developed, with arguably the most prevalent being Crosslinking and Immunoprecipitation followed by sequencing (CLIP-seq) [4] and its derivatives, such as PAR-CLIP [5], iCLIP [6] and eCLIP [7]. CLIP-seq data are commonly post-processed with peak callers, which identify, from the mapped reads, regions of enriched signal over background, i.e. binding sites. While experimental methods give an unprecedented insight into the binding specificities of RBPs, *in vivo* profiling of protein–RNA interactions is subject to the transcript abundances in the experimental cell type. Thus, researchers must instead rely on computational methods to impute missing binding sites on non-expressed transcripts or to characterize RBP-binding sites in settings where no experimental data are available, in order to avoid numerous costly experiments across a wide range of experimental conditions.

Method development for RBP binding-site prediction is an active area of research in the domain of computational RNA biology and an abundance of RBP binding-site prediction methods have been developed in recent years [8–10]. Development of new methods further accelerated with the advent of deep learning, which showed ground breaking performance improvements in many domains of research, including genomics. Current state-of-the art methods for RBP binding-site prediction are usually formulated as a supervised learning problem, to predict whether an RNA sequence is bound or not bound by a certain RBP. Bound regions are usually defined as high-confidence binding sites, so called *peaks*, from CLIP-seq experiments. Models are then trained to classify RNA sequences as bound or unbound, either in a single-task (one RBP per time) or in a multi-task (several RBP simultaneously) manner [11]. Given this rapid development of several predictive models (Figure 1A), it is becoming increasingly difficult for both experimental and computational RNA biologists to select the most appropriate method for the task at hand. This is largely due to the fact that studies train and evaluate their methods on different CLIP-seq datasets, which either encompass a different set of profiled RBPs or may contain binding sites that have been derived via different experimental CLIP-seq protocols (Tables 1 and 2). Indeed, it has been shown that different RBPs show a different degree of binding specificity [12] and thus, prediction methods have different upper and lower baselines, depending on the composition of the evaluation dataset. Further, CLIP-seq protocols differ in their signal footprint. For instance, protocols, such as iCLIP [6] or eCLIP [2], profile protein–RNA interaction at single-nucleotide resolution, raising the question whether an increase in predictive performance is due to an improvement in data quality, rather than an improvement of the computational methods. Classification methods require annotation of sequences with *positive* and *negative* labels, such

Received: January 31, 2023. **Revised:** June 15, 2023. **Accepted:** July 18, 2023

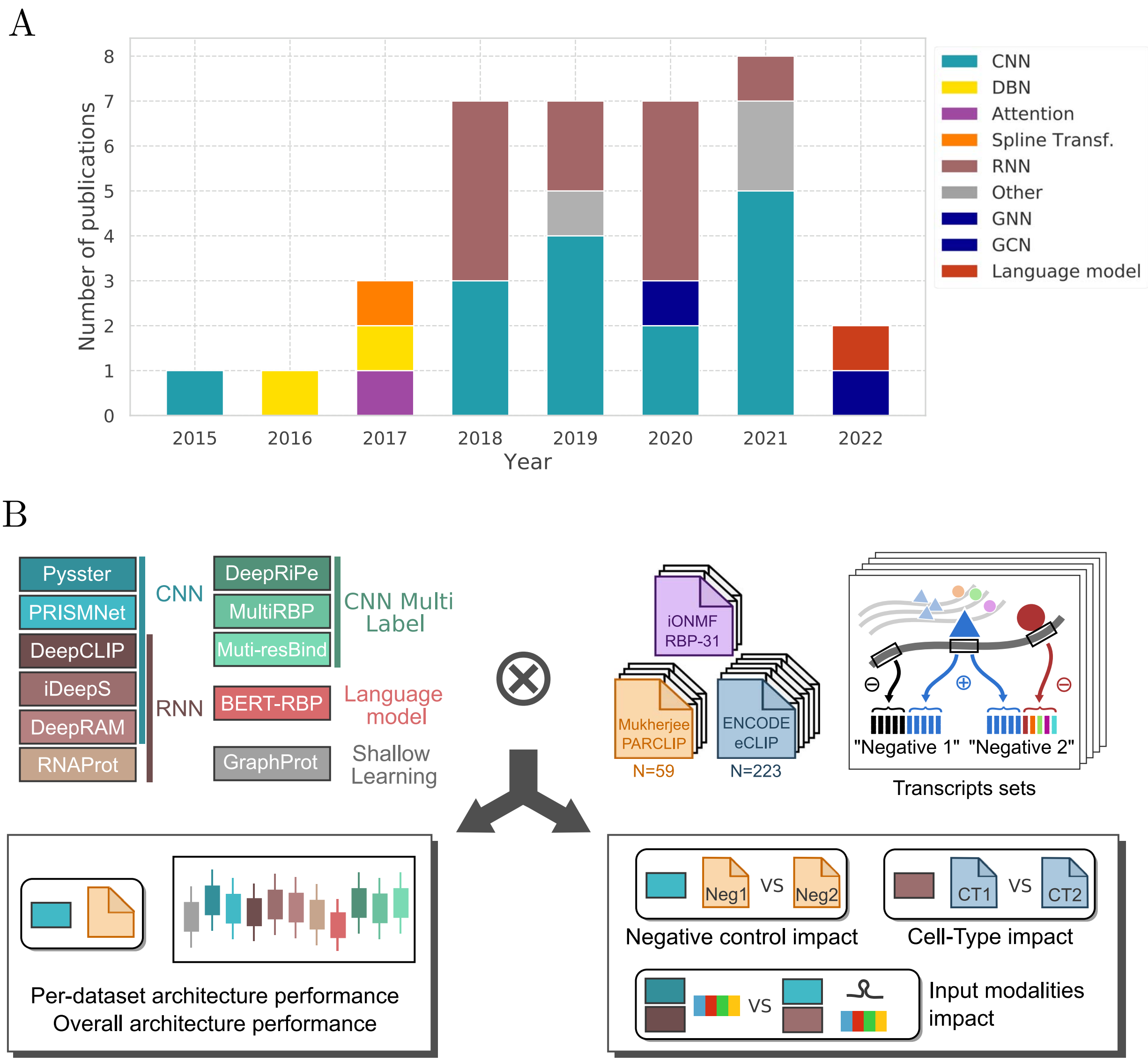


Figure 1. Overview of models and schematic of our benchmark. (**A**) Cumulative barplot of the published methods (see Table 1) representing the evolution of architecture choice over the years. (**B**) Illustration of the benchmark presented in this work. Methods representing various architectures were selected. Three datasets of experimentally derived binding sites were preprocessed into folds, separating sequences basing on their transcript assignment. From the selected models and datasets, we first perform an all-against-all evaluation to rank architectures and models (per dataset and across datasets). We also evaluate the impact of negative control sampling on model performance, as well as input modalities. Finally, we explore datasets properties, such as cell-type impact for matched RBPs.

that during training a decision function between the two classes can be learned. While CLIP-seq followed by peak calling explicitly yields a set of positive samples, negative samples are less trivial to obtain. Different negative sample generation strategies were developed across methods, further increasing heterogeneity during method evaluation. To date, multiple studies showed the presence of intrinsic biases in CLIP-seq, such as enhanced crosslinking likelihood at uridines, presence of stick-RBPs and RNase-bias towards termination at guanines [13, 14]. These biases may be predominantly present in the positive class set and may therefore serve as features for class-discrimination, leading to inflated method performances.

In this study, we describe 37 approaches and systematically benchmark 11 RBP binding-site prediction methods on binding sites of 3 large CLIP-seq repositories, comprising a total of 313 unique CLIP-seq experiments across 203 RBPs. Datasets are derived from (a combination of) common CLIP-seq protocols, including iCLIP [6], eCLIP [2] and PAR-CLIP [15]. We develop an uniform data pre-processing and training set construction strategy for all methods, enabling us to evaluate method performance in an unbiased way and allowing us to contrast methods exclusively based on method-intrinsic properties. Notably, we employ two negative-class generation strategies, where one strategy is agnostic to CLIP-seq biases while the other performs bias-aware sampling. We evaluate common properties of methods and their potential impact on model performance, with respect to architecture design choices and input modalities, such as the use of secondary RNA structure as an auxiliary input. Further, we perform cross-evaluation of models for two different cell types, K562 and HepG2, to address the question of whether models trained on CLIP-seq data from one cell type are suitable for prediction on another. Finally we investigate the potential generalization of

Table 1. Overview of methods dedicated to predicting the binding propensity of RNA-binding proteins for a given genomic interval of interest. Star symbol '∗' indicates shallow-learning methods. In red are the methods selected for the benchmark

Method	Year	Data type	Evidence	Architecture	Prediction	Ref
GraphProt∗	2014	RBP-24	Sequence structure	Graph embedding SVM	Binary	[20]
DeepBind	2015	RNAcompete	Sequence	CNN	Intensity	[45]
Deepnet-RBP	2016	RBP-24	Sequence structure 3D structure	DBN	Binary	[29]
iDeepA	2017	RBP-24	Sequence	Attention, CNN	Binary	[46]
Concise	2017	VanNostrand_ENCODE RBP-31	Sequence genomic annotation	spline transformation, CNN	Position-wise signal	[43]
iDeep	2017	RBP-31	Sequence structure genomic annotation motif, co-binding	DBN, CNN	Binary	[35]
iDeepS	2018	RBP-31	Sequence structure	CNN, BLSTM	Binary	[30]
pysster	2018	VanNostrand_ENCODE	Sequence	CNN	Binary	[28]
DLPRB	2018	RNAcompete	Sequence structure	CNN,RNN	Intensity	[33]
cDeepBind	2018	RNAcompete	Sequence structure	CNN, LSTM	Intensity	[27]
iDeepV	2018	RBP-67	Sequence	word2vec, CNN	Binary	[55]
iDeepE	2018	RBP-24	Sequence	CNN (global+local)	Binary	[47]
iDeepM	2018	RBP-67	Sequence	CNN, LSTM	Binary	[48]
DeepRAM / ECBLSTM	2019	RBP-31	Sequence	CNN, BLSTM, k-mer embedding	Binary	[49]
SeqWeaver	2019	Authors' datasets	Sequence	CNN	Binary	[80]
RDense	2019	RNAcompete	Sequence structure	CNN, BLSTM	Intensity	[32]
ThermoNet	2019	RNAcompete	Sequence structure	CNN	Intensity	[74]
mmCNN	2019	RBP-24	Sequence structure	CNN	Binary	[37]
iCapsule	2019	Authors' datasets	Sequence structure	Capsule Network	Binary	[26]
HOCNNLB	2019	RBP-31	Sequence	k-mer embedding CNN	Binary	[63]
DeepCLIP	2020	RBP-24;RNAcompete VanNostrand_ENCODE	Sequence	CNN, BLSTM	Binary	[50]
DeepRiPe	2020	Mukherjee_PARCLIP VanNostrand_ENCODE	Sequence genomic annotation	CNN	Multi-label probability	[40]
RPI-NET / RNAonGraph	2020	RBP-24	Sequence structure	GNN	Binary	[36]
DeepRKE	2020	RBP-24;RBP-31	Sequence structure	embedding, CNN, BLSTM	Binary	[25]
iDeepMV	2020	RBP-67	Sequence	Multi-view CNNs ensemble	Binary	[81]
DeepA-RBPBS	2020	RBP-31	Sequence structure	CNN, biGRU	Binary	[64]
MSC-GRU	2020	RBP-31	Sequence	CNN, biLSTM	Binary	[65]
MultiRBP	2021	RNAcompete	Sequence structure	CNN	Multi-label intensity	[34]
PRISMNet	2021	VanNostrand_ENCODE	Sequence structure	CNN	Binary	[51]
RNAprot	2021	RBP-24	Sequence genomic annotation conservation	LSTM	Binary	[42]
Multi-resBind	2021	Mukherjee_PARCLIP VanNostrand_ENCODE	Sequence genomic annotation	CNN	Multi-label probability	[41]
RBP-ADDA	2021	RNAcompete	Sequence	Adversarial domain adaptation	Binary	[82]
ResidualBind	2021	RNAcompete	Sequence	CNN	Intensity	[53]
kDeepBind	2021	RBP-31	Sequence	k-mer embedding CNN	Binary	[62]

(*Continued*)

Table 1. Continued

Method	Year	Data type	Evidence	Architecture	Prediction	Ref
RBPSpot	2021	ENCORI	Sequence structure	embedding DNN	Binary	[54]
BERT-RBP	2022	VanNostrand_ENCODE	Sequence	Language model	Binary	[59]
DeepPN	2022	RBP-24	Sequence	CNN, GCN	Binary	[83]

Table 2. Overview of datasets used by methods considered in this review. In bold are the datasets selected for the benchmark

Dataset	Year	First introduced by	Description and numbers	Ref
RNAcompete	2013	RNAcompete	207 RBPs, across 231 experiments. Total of 241 000 sequences each 38–41 nt length, split into 2 sets: set A (120,326) and set B (121,031). Each sequence has a score from the log odds ratio of intensities of pull-down vs input.	[44]
RBP-24	2014	GraphProt	Sites derived from CLIP-seq experiments. 24 experiments (23 from Dorina) for 21 RBPs. 16 PAR-CLIP, 4 HITS-CLIP, 4 iCLIP.	[20]
AURA2	2014	AURA2	Database of post-transcriptional regulatory interactions in UTR, including binding sites of RBPs.	[84]
RBP-31	2016	iONMF	Mix of PAR-CLIP, iCLIP and HITS-CLIP. 31 CLIP experiments for 19 RBPs. Positive controls: derived from positions with high read-counts. Negative controls: positions sampled from genes without interactions from any of the 19 RBPs.	[61]
RBP-67	2016	RNAcommender	67 distinct RBPs, 72 226 UTRs. Total of 502 178 interactions curated from the AURA2 database.	[85]
Mukherjee_PARCLIP	2019	DeepRiPe	PAR-CLIP experiments for 59 RBPs profiled in the HEK293 cell line.	[39]
VanNostrand_ENCODE	2020	Van Nostrand *et al.*	150 RBPs assessed in two cell-types (HepG2, K562), for a total of 223 experiments.	[7]

methods across CLIP-seq protocols, by performing cross-CLIP-seq evaluation where models trained from one protocol are applied onto data from other protocols.

The remainder of this article is structured as follows: First, we give a brief overview over machine learning methods for protein-RNA interaction prediction with a focus on input modalities and deep learning model architectures. Next, we cover benchmarking datasets and their preprocessing, before introducing methods selected for benchmarking. Third, we introduce our benchmark design, summarized in Figure 1B, including train/test splitting, negative sample generation and evaluation metrics. Finally, we report and critically discuss benchmarking results.

PREDICTING PROTEIN–RNA INTERACTION: FROM SHALLOW LEARNING TO DEEP LEARNING

Predicting protein-binding sites on arbitrary RNA or DNA sequences is a long-standing and unsolved task of computational biology research. While initial methods were predominantly mechanistic programs, often operating via scanning of sequences for a suitable binding site using a position-weight-matrix (PWM) representation of the protein's target sites [16–19], the field soon shifted towards more general machine learning methods, which allow for protein-binding prediction without first deriving an intermediate PWM representation. These methods were no longer constrained by the representation of binding preferences as fixed-length PWMs, which allowed for modeling of more complex protein–RNA interaction functions and resulted in greater predictive performance compared to classical approaches. For instance, GraphProt [20], a method based on a support vector machine (SVM), uses string and graph kernels to encode the primary and secondary structure of an RNA input. While traditional machine learning methods still relied on manual engineering of input features to classify RBP-bound versus unbound sequences, emergence of deep learning methods enabled quasi end-to-end model training. As a result, the research focus shifted from hand-crafting efficient representations of RNA sequence and auxiliary inputs, towards exploration of efficient deep learning architectures and informative input modalities.

In this study, following a comprehensive literature screening, 37 deep learning methods for the prediction of protein–RNA interaction *in vitro* and *in vivo* were identified. Methods were categorized based on their input modalities as well as neural network architecture elements and are summarized in Table 1.

Input modalities

The main input modality to predictive models of RNA–RBP binding sites is represented by the RNA primary sequence. Sequences of a fixed or variable length surrounding a potential RBP-binding sites are converted to either a numerical or one-hot-encoded representation and fed into the model. Besides RNA sequence as the primary input, models make use of a variety of auxiliary inputs, including RNA secondary and tertiary structure, genomic region context, evolutionary conservation and protein co-binding (Table 1).

RNA secondary and higher-order structure

The higher-order structure of an RNA sequence has been shown to play a vital role in facilitating interactions with the RNA molecule and proteins [21]. For instance, Roquin-1 binds to transcripts via recognition of specific stem loop structures [22] to regulate the post-transcriptional degradation of its targets. Several methods make use of computational RNA folding tools, such as RNAfold [23] or RNAshapes [24], to predict and incorporate secondary RNA structure as additional method inputs. These methods differ significantly with respect to how predicted structures are incorporated into the model. DeepRKE [25], iCapsule [26], cDeepBind [27], pysster [28], Deepnet-RBP [29] and iDeepS [30] first predict a minimum free-energy (MFE) structure via RNAshapes [24], before projecting each position in the input sequence onto an element from *structural vocabulary*, such as hairpin, multi-loop or stem. Subsequently, the structural vocabulary sequence is one-hot encoded. Notably, Deepnet-RBP additionally uses R3DMA [31] to additionally annotate hairpin and internal loop regions with probable tertiary structural motifs. RDense [32], DLPRB [33] and MultiRBP [34] refine this approach by computing the relative frequency of structural elements at each position via RNAplfold [23]. Here, each position in the input encodes secondary structure as a categorical distribution over structural elements. This represents an extension to one-hot encoding, which only takes into account the single MFE structure. Rather than considering structural elements, iDeep [35] uses RNAplfold [23] to predict the unpaired-probability of each position in the input sequence. RPINet [36] and mmCNN [37] operate directly on the predicted base-pairing probability matrix (BPPM). While mmCNN scans the BPPM via 2D convolutional operations, RPINet encodes base-pairing probabilities between input positions as weighted edges in a graph. Lastly, PrismNet is the only method that uses experimentally derived *in vivo* structure via icSHAPE [38], which yields a position-wise score across input sequencing, indicating whether a given position is paired or unpaired. The score vector is concatenated with the one-hot encoded RNA input before being passed to the model. While predicted structure is generally cell-type agnostic, as the same RNA structure is used for prediction, icSHAPE provides cell-type specific structure information. Given that RBPs may exhibit cell-type specific binding [7], this may represent an advantage over structure prediction methods. Further, methods only incorporate information on the predicted minimum free-energy structure (such as iDeepS or DeepRKE) may lack information on other viable conformations in the RNA's secondary structure ensemble, which may only have marginally higher free-energy values. On the other hand, icSHAPE data show a high degree of sparsity and does not provide proper folding information, but instead probes, which nucleotides are paired or unpaired.

Genomic context

Analysis of transcriptome-wide binding site locations showed that many RBPs preferentially bind to specific genomic regions or landmarks [39]. For instance, splicing-associated RBPs predominantly bind at splice junctions and thus information on whether an input sequence is derived from exons, introns or lies at their junction may serve as additional evidence for protein–RNA interaction prediction. DeepRiPe [40], Multi-resBind [41] and iDeep [35] use a region-vocabulary approach to encode the genomic context of a given input sequence. Here, input positions are first annotated with one of 4 (5 in case of iDeep) genomic region types, including CDS, 5'/3' UTR, introns and exons, followed by one-hot encoding. Using a similar approach, RNAProt [42] maps positions to exon/intron regions, prior to one-hot encoding. Concise [43] computes the distance of each input position to a set of genomic landmarks, including 5' splice sites, poly-A sites and transcription start sites. Raw distances are transformed to smoothed representations using spline transformations.

Sequence conservation, co-binding and motifs

Interaction with RBPs determines the fate of transcripts and thus, disruption of RBP target sites may lead to misregulation of post-transcriptional processes and disease. Therefore, RBP target sites are expected to show a higher degree of evolutionary conservation when compared to non-target sites. Leveraging this fact, RNAProt [42] incorporates phastCons and phyloP conservation scores as auxiliary inputs. To leverage prior knowledge, iDeep [35] compute motif-scores for 102 human RBPs using the CISBP-RNA [44] database. Input sequences are additionally annotated with co-binding information using experimental data from other RBPs.

Deep learning architectures for protein–RNA interaction prediction

DeepBind [45] was one of the first methods which employed deep learning for the prediction of protein-binding from nucleotide sequences, demonstrating ground-break on both *in vitro* and *in vivo* protein–RNA interaction datasets. DeepBind makes use of a single 1D Convolutional layer, which consists of a set of short, learnable filters that are applied over the input sequence. Over the course of training, the filter-weights are adjusted to yield high activation scores at sequence locations which represent potential binding targets, loosely resembling PWMs scanning of classical methods. Commonly, the outputs of convolutional filters serve as input to downstream layers, such as additional convolutional or recurrent layers, or are directly fed into a linear classifier, which predicts the final binding affinity of the protein of interest for the given RNA sequence. Convolutional neural networks (CNNs) are at the heart of several protein–RNA predictions methods, including iDeep{A,S,E,M} [30, 46–48], pysster [28], DeepRAM [49], DeepCLIP [50], DeepRiPe [40], MultiRBP [34], PrismNet [51] and Multi-resBind [41], among others (see Table 1). Pysster [28] increases the number of convolutional layers to 3, resulting in a deeper model which can potentially learn more complex binding functions. An additional increase of the input length to 400 (from 101 in case of DeepBind) further increases the receptive field of the model and allows it to consider a broader sequence context around potential binding sites. DeepRiPe [40] uses a single convolutional layer and jointly predicts binding of several RBPs in a multi-task manner, exploiting the fact that many RBPs shared binding preferences and tend to co-bind. This results in an efficient model representation, as convolutional filters are shared across tasks (RBPs), potentially increasing model performance and training stability. Notably, DeepRiPe scans both sequence and genomic region (Section 2.1.2, Genomic Context) inputs with separate CNN modules, before joining their outputs for the final classification. A similar approach is used by iDeepS [30] for the independent processing of sequence and secondary structure inputs. PrismNet [51] and Multi-resBind [41] (another multi-task model) further increase network depth via stacking of convolutional layers, while adding residual connections to combat the vanishing gradient problem. PrismNet [51] additionally makes use of a squeeze-and-excitation (SE) module [52] to recalibrate outputs of the first convolutional layer. In contrast to DeepRiPe and Multi-resBind, the multi-task method MultiRBP [34] uses convolutional kernels of varying size in the same layer, in order to accommodate binding footprints of different size, across a large number of RBPs. To increase the receptive field of the CNN model, ResidualBind [53] uses several dilated convolutional layers with exponentially increasing dilation coefficient.

Convolutional filters act as (partial) motif detectors, where more complex binding motifs are constructed from the outputs of previous convolutional layers, as the network depth increases. To aggregate the outputs of convolutional layers across the entire sequence, multiple methods make use of recurrent layers, such as long-short-term-memory (LSTM) or gated-recurrent-units (GRU), which enable efficient learning of long-range dependencies. DeepCLIP [50], cDeepBind [27], iDeepS [30], deepRAM's ECBLSTM model [49] and DeepRKE [25] use a bidirectional LSTM (BLSTM) on top of the outputs of the preceding convolutional layer. While in case of iDeepS, cDeepBind, deepRAM and DeepRKE, the BLSTM output serves as input to a final linear classifier, DeepCLIP directly returns the sum over the BLSTM output as a binding affinity score. Other methods are based on an exclusively recurrent architecture, such as RNAProt [42], which uses an LSTM that directly operates on the RNA sequence. DLPRB [33] combines convolutional and recurrent layers in an alternative way, by concatenating the outputs of a convolutional and a recurrent module, both operating on the RNA sequence, which are then fed into a final linear classifier. The majority of methods use one-hot encoding to project the RNA sequence into a machine-readable format. However, several methods, including RBPSpot [54], iDeepV [55] and deepRAM [49], use a word2vec [56] model to first learn an embedding of nucleotide 3-mers in an unsupervised manner. During training, k-mers of the input RNA sequence are projected into the word2vec model's embedding, which then serves as input to subsequent layers. Recently, Transformer-based models emerged as an alternative architecture to CNN- and RNN-based models in fields of natural language processing (NLP) as well as computational biology research [57, 58]. Pre-trained on large corpora of unlabeled data, these models showed ground-breaking performance when fine-tuned on task specific, labeled data. BERT-RBP [59] uses a DNABERT [60] model, pre-trained on a tokenized version of the human genome and fine-tunes it on *in vivo* protein–RNA interaction data. With over 100 million trainable parameters, BERT-RBP represents the largest-capacity deep learning model for protein–RNA interaction prediction evaluated in this study by a large margin. To jointly incorporate RNA sequence and secondary structure graph representations (Section 2.1.1, RNA Secondary and Higher-Order Structure), RPINet [36] uses a modified graph convolutional network (GCN). In each layer, the current node embedding is updated via a graph convolutional operation on the predicted BPP matrix and a convolutional operation along the sequence axis. To obtain binary predictions, the final input embedding is processed by a LSTM layer.

MATERIAL AND METHODS

Data and preprocessing

RBP-binding prediction methods were trained and evaluated on binding sites from three distinct sets of experiments, derived from common CLIP-seq protocols, including eCLIP [2], PAR-CLIP [15] and iCLIP [6]. While these datasets were used as training and evaluation sets for some of the benchmarked methods in this study, no study systematically evaluated their method on all three datasets. The RBP / dataset matrix is shown in Table 2. Supplementary Table 1 provides a full list of all CLIP-seq experiments, including RBP, cell type and protocol.

ENCODE (eCLIP)

The ENCODE Project [7] contains the largest collection of CLIP-seq datasets to date, encompassing 223 eCLIP [2] experiments for 150 RBPs across two cell lines, HepG2 and K562. It has been utilized by a number of studies for model training and evaluation, including Pysster [28], DeepRiPe [40], DeepCLIP [50], Concise [43], Multi-resBind [41] and PrismNet [51]. For each experiment, a set of high-confidence peaks was obtained by processing the narrow-peaks BED files as follows. First, peaks of both replicates are intersected with transcripts of GENCODE (Version 42) to remove all peaks outside of transcript regions. The remaining peaks of both replicates are then intersected and peaks which are present in only one replicate are discarded. For each peak, we define the base at its 5' end as the single-nucleotide site of crosslinking between RBP and RNA, as suggested by Dominguez *et al.* [21]. To reduce the computational burden of the benchmark analysis and to select a set of high-quality cross-linked sites, we select at most the top 20 000 peaks with highest signal fold-change over the size-matched input (SMInput) for each experiment.

iONMF (PAR-CLIP, iCLIP, CLIP, HITS-CLIP)

The iONMF dataset was established by Stražar *et al.* [61] and has since been used by a number of methods for training and evaluation, including iDeep [35], iDeepS [30], DeepRAM [49], DeepRKE [25], kDeepBind [62], HOCNNLB [63], DeepA-RBPBS [64] and MSC-GRU [65]. It consists of cross-linked sites extracted from 31 CLIP-seq experiments for 19 RBPs and, in contrast to the ENCODE and Mukherjee et al. [39] datasets, it includes data derived from different CLIP-seq protocols, including PAR-CLIP [15], iCLIP [6] and HITS-CLIP [66]. The authors retrieved counts data from the iCount [67] and DoRiNA [68] database and selected, for each experiment, the top 100 000 nucleotide positions with the highest cDNA counts. For positions with a distance of <15, only the position with the highest cRNA count was considered while all other positions were ignored, as suggested by König *et al.* [6]. The authors then sampled at most 10 000 cross-linked sites for each experiment in order to reduce processing time. We obtain train and test sets of the iONMF dataset from github.com/mstrazar/iONMF. After merging of both sets, we discard all negative samples defined as part of the iONMF study to obtain the final set of positive cross-linked sites and perform no further processing.

Mukherjee et al. (PAR-CLIP)

This dataset is a subset of 59 PAR-CLIP experiments in the HEK293 cell line, aggregated from different studies and processed by Mukherjee *et al.* [39]. It was first used by Ghanbari *et al.* [40] for training and evaluation of DeepRiPe. Variable-length PARalyzer [69] peak regions in BED format were obtained for each PAR-CLIP experiment and peaks were centered to a single nucleotide 'pseudo-crosslink' position, in order to homogenize binding sites with the other datasets. Following Ghanbari *et al.* [40], we did not lift over genomic positions from GRCh37 to GRCh38, but instead used genome and GENCODE [70] versions for the GRCh37 assembly for all downstream processing of this dataset.

Protein–RNA interaction prediction methods

Among the 37 methods compiled in Table 1, we selected 10 deep learning methods for benchmarking, spanning a wide variety of model architectures and input modalities. As a point of reference for the performance of deep learning methods when compared to traditional machine learning approaches, we included GraphProt

[20], a shallow learning method based on SVM. Generally, methods were trained with hyperparameters specified in the original publications and otherwise default parameters, unless otherwise noted in the method specific paragraphs. For methods which set a side a validation set and monitor the validation loss for the purpose of early stopping, a uniform training/validation split of 80/20 was used, as this represents the majority split ratio among methods. Thus, models were trained for different number of epochs, according to the author's recommendations or the early-stopping criteria.

GraphProt (Steffen et al. 2014)

GraphProt [20] makes use of a SVM together with string and graph kernels to incorporate both sequence and predicted secondary structure information for classification of a given RNA sequence as bound/unbound. Specifically, during training, GraphProt selects CLIP-seq peaks of at most 75nt in size, which are extended by 15nt up-and down-stream to yield the model's *viewpoint*. To improve the quality of secondary structure predictions, the viewpoint is further extended by 150nt up-and down-stream, followed by prediction of the minimum free energy structure via RNAshapes [24]. Prior to feature extraction from the secondary structure graph via an extension of the NSPD kernel [71], additional information on the type of substructures (e.g. stem or hairpin-loop) is added via a hypergraph. Finally, SVM classification is performed on the basis of extracted sequence and graph features. Note that, as samples in our benchmark dataset are represented by single-nucleotide positions in the transcriptome as a way to homogenize preprocessing across datasets, we extended each sample upstream and downstream to the maximum allowed length of 75 + 15 nt to create the sample's viewpoint. This is followed by a bi-directional viewpoint extension of 150 nt for structure prediction. As GraphProt is an SVM classifier, it does not output positive-class probabilities. To compute auROC performance scores, the classification margin of each test sample (i.e. the distance of the sample to the decision boundary) is used instead.

iDeepS (Pan et al. 2018)

iDeepS [30] takes as input an RNA sequence of 101nt and integrates both sequences and predicted secondary structure information via a bi-modal neural network architecture. The minimum free-energy secondary structure is predicted with RNAshapes [24] to yield a 6-symbol structure alphabet, indicating whether a given position in the input sequence resides in a stem (S), multiloop (M), hairpin (H), internal loop (I), dangling end (T) or dangling start (F). Sequence and structure are one-hot encoded and scanned independently by a single CNN layer. Feature maps of both CNN layers are merged and subsequently scanned by a bi-directional LSTM. Finally, the output feature map is passed through a one-unit linear layer with sigmoid activation, to predict the binding probability of the RBP on a given input sequence.

Pysster (Budach et al. 2018)

Pysster [28] is a Python framework for creating CNN models for genomic sequence-based classification and regression tasks. While Pysster may incorporate additional input features, such as secondary structure or genomic region information, Budach *et al.* [72] showed that high protein–RNA interaction prediction performance can be achieved using models trained on RNA-sequence alone. Pysster [28] takes as input a 400nt window and scans the one-hot encoded RNA sequence via a stack of *three* convolutional layers. The resulting feature maps then serves as input to a stack of two fully-connected layers. In contrast to other methods, Pysster defines two types of negatives, with one half being sampled uniformly from regions with no overlapping binding sites of the given RBP and the other half being sampled from binding sites of other RBPs in order to combat CLIP-seq specific cross-linking bias. The final output layer then consists of three units (one for the positive class and two for both negative classes) and a softmax activation, assigning predicted probabilities to the three classes. Note that in order to make Pysster predictions comparable to other classification methods, only two (positive and negative) output classes were used in this study and Pysster models were trained on just one type of negative samples. Pysster was originally trained using early-stopping by monitoring the validation loss on a 85/15 train/validation split for up to 200 epochs with a patience of 15. To harmonize the training/validation split ratio with other methods that employ early-stopping, we instead used a split ratio of 80/20. Following Budach *et al.* [72], Pysster was trained using 3 CNN layers, each with 150 filters and a kernel size of 18 and otherwise default parameters. For training of Pysster with 101nt inputs, the kernel size was slightly reduced to 14, in order to avoid negative dimension sizes during convolution, as Pysster uses valid padding.

DeepRAM (Trabelsi et al. 2019)

DeepRAM [49] is a tool for building flexible deep learning architectures for binary classification of RNA and DNA sequences. Evaluated against both protein–DNA and protein–RNA interaction prediction datasets, the authors compared several common architectures, including combinations of convolutional, recurrent and embedding layers with respect to their predictive performance on the protein–RNA interaction prediction task. For this benchmark, we selected the best performing architecture, termed *ECBLSTM*, which consists of an RNA sequence 3-mer embedding, followed by a convolutional and bi-directional LSTM layer. As the authors evaluated their ECBLSTM model on the iONMF dataset (Stražar *et al.* [61]) with a sequence length of 101nt, we generated inputs of similar length across all benchmark datasets. DeepRAM first performs hyperparameter selection by training 40 models and evaluating their performance on 3-fold cross-validation, before selecting the best performing hyperparameters for training of a final model. As training 40 models across all benchmark datasets was computationally infeasible, we reduced the number of sampled hyperparameters to 5 during the benchmarking of deepRAM's ECBLSTM architecture. Further, the authors additionally train 5 models using the best hyperparameter configuration on the full set of samples and return the model with the lowest training loss as the final model. This is not expected to significantly impact generalization performance, we omitted this step by training only a single model on the optimal hyperparameter configuration, in order to further improve the runtime of deepRAM. Besides those changes, deepRAM was trained as described by Trabelsi et al. [49].

DeepRiPe (Ghanbari et al. 2020)

DeepRiPe [40] is a multi-label classifier which operates on two input modalities, sequence and genomic annotation, with the latter consisting of a mapping of each position in the input RNA sequence to one of four genomic regions, CDS, intron, 3' UTR and 5' UTR. Sequence and genomic region inputs are one-hot encoded, before being processed independently by a convolutional layer. The feature maps of both layers are subsequently concatenated and processed by a CNN or GRU layer, before being passed to a fully connected and output layer for classification. DeepRiPe is trained on non-overlapping windows from the human transcriptome, which are labeled with one or more RBPs. Crucially, this alleviates the need for defining negative samples explicitly during training, as windows which are positives for a set of RBPs serve as negatives for the rest. DeepRiPe bins RBPs based on the number of observed binding sites in the corresponding PAR-CLIP or eCLIP experiment and trains multiple models, one for each bin. Further, input sizes of 50nt and 150nt are used for PAR-CLIP and eCLIP samples, respectively. Here, we instead train a single DeepRiPe model with a RNA sequence input size of 150nt and 250nt for the genomic region input, for each of the three datasets, to make processing consistent across datasets. In contrast to the authors, we also did not remove experiments with less than 1000 binding sites, to enable direct comparison with other methods. The authors trained models on 80% of the training data, while 10% were used for validation and testing, respectively. Here, a 80/20 train/validation split was performed on our generated training samples (Section 3.3.1, Transcript-Level Training/Test Splitting) and models were trained for at most 40 epochs and early stopping with a patience of 5.

DeepCLIP (Gronning et al. 2020)

DeepCLIP [50] is a sequence-only classifier, which first one-hot encodes an input RNA sequence, before applying a convolutional layer as a motif extractor. The resulting feature map is subsequently fed into a bi-directional LSTM layer to perform binary classification of RNA sequences. Notably, in addition to classification, DeepCLIP yields a single-nucleotide binding profile along the input sequence, separating it from the majority of tools for protein–RNA interaction prediction, which usually exclusively perform binary classification of a given input sequence. Similar to GraphProt [20] and RNAProt [42], DeepCLIP takes variable-length input sequence and was evaluated on binding site regions of 12–75nt. In this benchmark, all inputs were length-normalized to a 75nt window centered around the RBP-binding site. Further, the authors trained DeepCLIP for a varying number of epochs together with early stopping, depending on the size of the given training dataset. As no specific guidelines for determining the maximum number of training epochs as well as the early stopping patience are provided by the authors, the most prominent choice of $max\_epochs = 200$ and $patience = 20$ from the publication [50] is used for benchmarking DeepCLIP in this study. Again, we harmonized the training/validation split ratio with other methods that employ early-stopping by choosing a split ratio of 80/20.

PrismNet (Sun et al. 2021)

PrismNet [51] is the first method to incorporate RNA sequence and experimental structure data measured with *in vivo* click selective 2'-hydroxylacylation and profiling experiment (icSHAPE) [38] to predict RBP binding. icSHAPE assigns reactivity scores at transcriptome-wide level and nucleotide resolution. These scores range between 0 and 1, with the lower scores indicating less reactivity and therefore likely representing paired positions. Sun *et al.* generated icSHAPE data for seven different cell lines, including HepG2, K562 and HEK293, covering both the ENCODE and Mukherjee datasets. The input to the model is a one-hot encoding of a 101nt input RNA sequence, to which the secondary structure icSHAPE-score vector is appended. The concatenated input is fed into a neural network consisting of a squeeze-and-excitation (SE) module and multiple convolutional layers with residual connections. A fully-connected layer then performs the final binary classification of the input to bound/unbound. We obtained icSHAPE data from Sun et al. [51], lifting over coordinates from GRCh38 to GRCh37 for the Mukherjee and iONMF datasets, using UCSC's LiftOver tool. As no matching icSHAPE was available for the U266 cell line in the iONMF dataset, icSHAPE vectors were defaulted to −1.0, to indicate missing values. As only two experiments were affected (IDs 19 and 20), benchmark evaluation results are not expected to be affected significantly. Training was performed for 200 epochs using early stopping with a patience of 20 and a training/validation split of 80/20.

MultiRBP (Karin et al. 2021)

MultiRBP [34] is trained on RNAcompete [73] and subsequently evaluated on eCLIP data. It can be trained on sequence as well as structure, but was demonstrated to performed best when being trained only on the former with an input sequence length of 75 nucleotides. Similar to DeepRiPe [40] and Multi-resBind [41], MultiRBP is a multi-task model which predicts binding affinities for multiple RBPs at once. The CNN-based architecture involves two different branches with varying filter size, which are concatenated just before the output layer, in analogy to ThermoNet [74]. Both include global max-pooling, fully-connected layers and convolutional layers with kernels of varying size. Most notably, since training is done on *in vitro* data, the model is trained on predicting scalar-binding intensities rather than binary labels and evaluated on *in vivo* classification without adaptation of the model output. Here, we trained MultiRBP without changing the implementation of the model apart from the size of the output vector, in order to match the amount of RBPs in the three datasets. Data was preprocessed in analogy to DeepRiPe, but with an input size of 75nt. As described by the authors, training was performed for 78 epochs without early stopping on a validation set.

RNAProt (Uhl et al. 2021)

RNAProt [42] is a toolkit for RBP-binding sites prediction, integrating a set of utilities from training-dataset generation to reporting statistics and visual information such as logos of extracted motifs. The model is based on RNNs and in its basic configuration takes as input a 81nt RNA sequence, allowing the user to optionally provide additional features, including secondary structure information, conservation scores, exon-intron annotation, transcript region and repeat region annotation. Here we used the default architecture variant (RNN–GRU) and trained the model with a configuration that was performing best according to the benchmark provided by the authors. This configuration makes use of sequence, exon–intron annotation and phyloP [75] and phastCons [76] conservation scores computed on alignments of 100 vertebrates. Following the authors, the model was trained for a maximum of 200 epochs, with early stopping set at 30, and a train-validation split of 80/20.

BERT-RBP (Yamada et al. 2021)

BERT-RBP [59] is a sequence-based model that makes use of DNABERT [60], a large nucleotide language model pre-trained on the human genome, which is fine-tuned to perform protein–RNA interaction binary classification. With over 100 million trainable parameters, BERT-RBP represents the largest model evaluated in this benchmark. Given a 101nt input RNA sequence, the sequence is first tokenized into overlapping 3-mer nucleotides. Tokens are embedded and then fed into a 12 layer transformer encoder. The final encoding of the *CLS* token, prepended to the RNA sequence input, is then passed to a classifier, in order to predict binding/non-binding. Following instructions in the authors Supplementary Table S1, BERT-RBP was trained for 5 epochs. Initial training and evaluation of BERT-RBP with default parameters showed high instability during training (Supplementary Figure 2).

After consulting Yamada *et al.* [59], BERT-RBP was retrained BERT-RBP with an updated set of hyperparameters, namely an increased batch size of 256, a lower learning rate of $2e^{-5}$ and an increased number of epochs of 20. To prevent the model from over-fitting due to an increased number of epochs, we performed a train-validation split of 80/20 and added the *–evaluate_during_training –early_stop* 2 flags to training runs, as suggested by the authors. Main results on BERT-RBP are obtained from training runs on this modified set of parameters.

Multi-resBind (Zhao et al. 2021)

Multi-resBind [41], similarly to DeepRiPe, trains a multi-task model on CLIP data, but uses a deeper architecture, adding more convolutional layers and residual skip connections. The model was trained on all possible combinations of sequence, structure and region features, but performed best with only 150nt long sequence and region features as a concatenated input vector. We used the exact same preprocessing routine as in DeepRiPe with the exception that we extracted region features with a size of 150 rather than 250. This was done since Multi-resBind, as opposed to DeepRiPe, is a single input model and requires all input features to have the same size. Further steps were done like in the publication, training the model for 40 epochs and evaluating the one with the best validation loss. As done for DeepRiPe, we trained one model on the entire dataset rather 3 different ones as done in the paper in order to keep the training consistent across methods.

Benchmark design

Transcript-level training/test splitting

Machine-and Deep-Learning methods require training-and test-sets for the parameter learning and subsequent estimation of the model generalization performance. Crucially, those sets must not intersect in order to avoid over-estimation of the model performance. To this end, several methods randomly split binding sites into training- and test-sets, which may violate the empty-intersection requirement, as peak callers such as CLIPper [77] or PARalyzer [69] can produce overlapping peak regions. Consequently, randomly splitting binding sites into training- and test-sets may lead to over-estimation of model performance.

To ensure that training- and test-sets do not intersect, we employ a transcript-level approach by first dividing human coding and non-coding transcripts into equally-sized, non-overlapping sets and subsequently assigning binding sites to each set. Transcript regions were gathered from GENCODE Version 42, for both the GRCh38 and GRCh37 assemblies, and transcripts overlapping on the same strand were merged and subsequently split into 5 equally-sized sets. Binding sites of each experiment are then intersected with merged transcripts, thus assigning them to one of the 5 sets, such that each set contains roughly 20% of an experiment's binding sites. While binding sites directly serve as positive-class instances, negative-class instances for a given set were generated exclusively using (merged) transcripts within the set (as described in Section 3.3.2, Generation of Negative-Class Samples), in order to prevent data-leakage between sets. Evaluation of generalization performance was then performed on the first set, while training was performed on the union of samples of the remaining sets. Notably, this approach allows for a 5-fold cross-validation evaluation of methods, which was omitted due to the computational burden associated with training and evaluation of a large number of deep learning models.

Generation of negative-class samples

Machine learning methods for binary classification require both positive and negative sample instances. However, through CLIP-seq experiments, only positive (i.e. cross-linking) events are observed explicitly, while negatives (i.e. no cross-linking) events need to be defined implicitly, for instance via the absence of observed binding. Several methods, including DeepBind [29], iDeepS [30] and PrismNet [51], sample negatives uniformly from transcriptome regions not overlapping with observed binding sites. This assumes that under absence of an RBP-specific binding features, identifying cross-linked positions is equally (un-)likely across all transcriptome positions. Some studies, including RNAProt [42] and DeepCLIP [50], refine this approach by restricting sampling of negatives to transcripts harboring at least one observed binding site. This ensures that the negatives are derived from transcripts expressed in the experimental cell type and therefore present as a binding partner for the RBP of interest at the time of the experiment, constraining the previous assumption. Recent studies suggest that CLIP-seq experiments suffer from several technical biases, such as background signal, highly abundant RNA, enhanced photoreactivity of uridines or library contamination with RNA fragments of other RBPs [13, 14, 78]. Models trained with negative instances sampled uniformly from unbound regions of (expressed) transcripts are prone to incorporate these biases in their learned function, as CLIP-seq biases are expected to be predominantly present in positive instances. Thus, these models may only partially predict true protein–RNA interaction and performance estimates may therefore be overly optimistic.

To combat this, different strategies have been developed in order to prevent models from learning uninformative biases. For instance, Pysster [28] supplements its set of negatives with cross-linked sites of *other* RBPs, while DeepRiPe [40], a multi-task method, eliminates the need of explicit negatives altogether, as the positive-class instance of one RBP may serve as a negative-class instance of another RBP during model training. To compare the performance of different methods in an unbiased and fair manner, we employed the same negative sample-generation strategy for all methods. Throughout this study, methods are trained and evaluated using two negative-class generation strategies. We construct a set of negatives for each experiment by uniformly sampling positions from transcripts overlapping with at least one binding site of the protein of interest, hereafter referred to as *negative-1*. A second set of negatives is constructed by sampling from binding sites of *other* RBPs experimentally assessed in a given dataset. This ensures that CLIP-seq biases are equally present in the positive and negative set, which renders CLIP-seq biases uninformative with respect to positive/negative class separation and thus prevents learning of CLIP-seq biases during model training. This negative set, consisting of positives of other RBPs, is hereafter referred to as *negative-2*. To prevent sequences from being present in both the negative and positive set, for instance due to co-binding of RBPs, we only sample positives of other RBPs which do not overlap with positives of the target RBP. For both strategies, negatives were generated at a ratio of 1:1 with respect to the number of positive-class instances for each experiment and training and evaluation of all methods was performed separately on both sets of negatives.

Generating method inputs

Unless otherwise noted (Section 3.2, Protein-RNA Interaction Prediction Methods), we construct method inputs as described by the respective authors.

Performance evaluation metrics

Method classification performances are reported as the area under the receiver operating curve (AUROC) and precision-recall

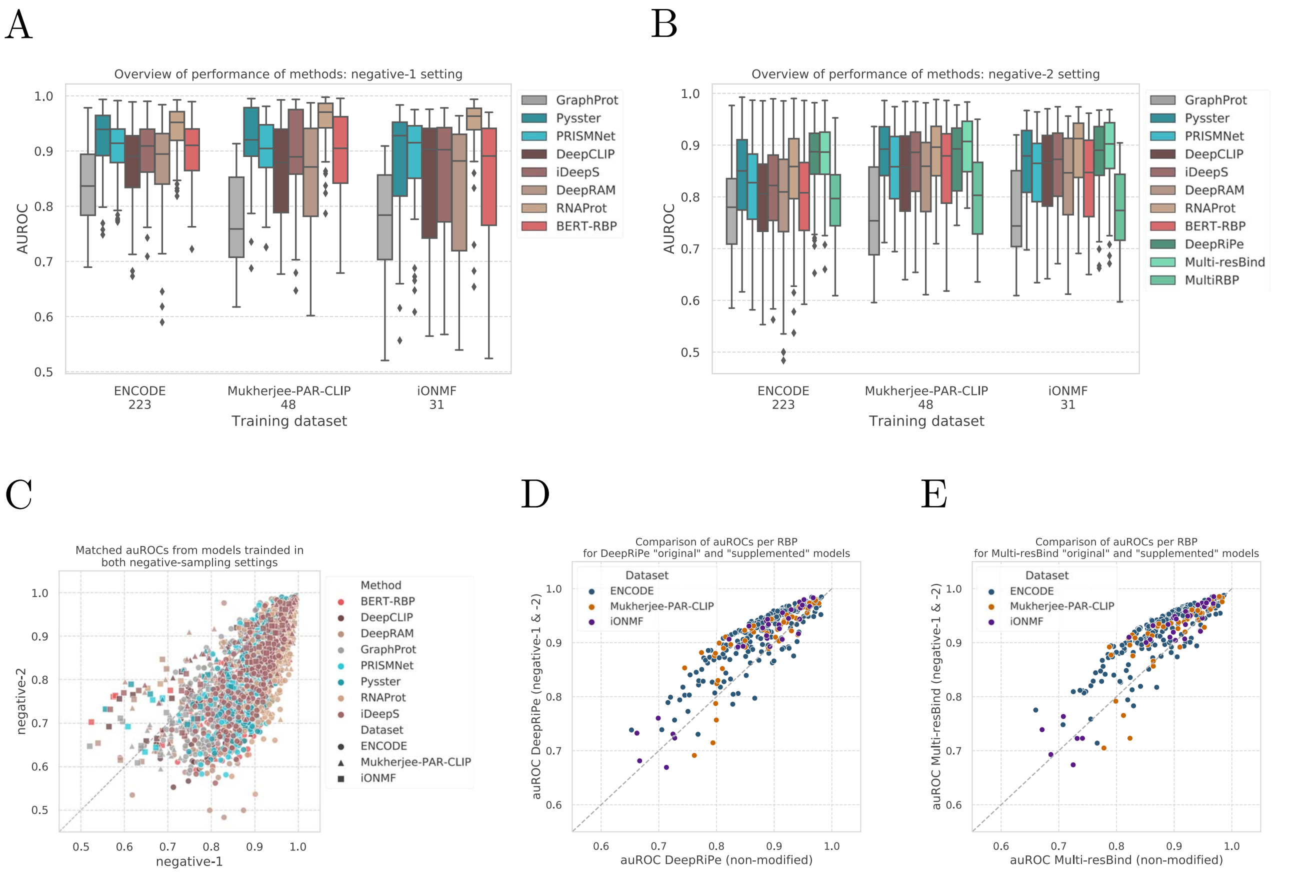


Figure 2. Results from the Benchmark. (**A** and **B**) Distribution of auROC values of trained models, across datasets, under the negative-1 (A) and negative-2 (B) sampling schemes. Multi-label models are absent from negative-1 setting due to not handling such negative samples. (**C**) Comparison of auROCs from models trained under the two schemes of negative control sampling. (**D** and **E**) Comparison of the classification performance of the modified multi-label methods (enabling negative-1 classification) with the original method, pairing classification results for each RBP, for DeepRiPe in (D) and Multi-resBind (E).

curve (AUPR) as well as the F1 score. As we are benchmarking a wide range of classifiers, including single- and multi-task classifiers, a drawback of the auPR and F1 score metrics is that their baseline performance, i.e. the expected performance of a uninformative random classifier, is subject to the class frequency. These metrics set multi-label classifiers (such as DeepRiPe [40], MultiRBP [34] and Multi-resBind [41]) at a disadvantage, as the frequency of a given label l is among n samples is $\frac{1}{n}\sum_i^n I(y_i = l)$ (where I is an indicator function) and thus much lower than the positive class frequency of 0.5 in the binary case. Therefore, to evaluate the discriminative power of methods in an unbiased manner, AUROC scores were used as the primary evaluation metric in this study.

RESULTS AND DISCUSSION

We trained a total of 4902 models across a matrix of 313 CLIP-seq experiments and 11 methods in each of the two (negative-1 and negative-2) settings. This includes one shallow learning, 2 and 3 CNN-based and RNN-based binary classification models, respectively, as well as 3 CNN-based multi-task methods (Figure 1).

Deep learning outperforms shallow learning methods

Figure 2a shows the AUROC of binary classification methods for models trained on positive and negative-1 samples. Performance of multi-task models (DeepRiPe, Multi-resBind and MultiRBP) is not displayed here, as these methods are trained without universal negative sequences by design. We observed no convergence in 1 deepRAM and 89 BERT-RBP models during training, leading to misbehavior during inference (predicting a single score for all samples) or random baseline performance (Supplementary Figure 2). These models were removed from the downstream evaluation. Following suggestions from Yamada *et al.* [59], we modified BERT-RBP hyperparameters before re-training it on all datasets (Methods). Note that all BERT-RBP performances in Figure 2 are reported with respect to the modified BERT-RBP models. Consequently, we disregard evaluation results of BERT-RBP in subsequent analysis. GraphProt, the only evaluated shallow learning method, showed the lowest performance with a mean auROC of 0.8211, followed by DeepRAM (0.8697), DeepCLIP (0.8733), BERT-RBP (0.8927), iDeepS (0.8932), PrismNet (0.9015), Pysster (0.9190), while RNAProt yielded the highest performance (0.9419) among binary classification methods (Table 3). Figure 2B shows auROC performance in the negative-2 setting. Here, multi-task methods are included, as by design the positives of one RBP may serve as negatives for another. We observe a strong decrease in performance for binary classification method compared to the negative-1 setting (Figure 2C), with a mean AUROC decrease of –0.0815 (BERT-RBP), –0.0760 (PrismNet), –0.0693 (Pysster), –0.0668 (DeepRAM), –0.0652 (DeepCLIP), –0.0647 (iDeepS) and –0.0850 (RNAProt) As outlined in Section 3.3.2 (Generation of Negative-Class Samples), we hypothesize that this may be due to CLIP-seq experimental biases, which, in case of the negative-1 setting, are exclusively present in the positive

Table 3. Performance of methods as measured by the average AUROC from each of the negative-control settings

Method	Mean AUROC negative-1	Mean AUROC negative-2	Delta AUROC
BERT-RBP	0.8927	0.8112	−0.0815
Pysster-101	0.9003	0.8311	−0.0692
PRISMNet	0.9015	0.8255	−0.0760
Pysster	0.9190	0.8497	−0.0693
DeepRAM	0.8697	0.8029	−0.0668
DeepCLIP	0.8733	0.8081	−0.0652
iDeepS	0.8932	0.8285	−0.0647
RNAProt-Extended	0.9419	0.8569	−0.0850
GraphProt	0.8211	0.7756	−0.0455
DeepRiPe	0.9103	0.8774	−0.0329
Multi-resBind	0.9174	0.8825	−0.0349
MultiRBP	NA	0.7937	NA

sample set, thus serving as a feature for class discrimination. In the negative-2 setting, biases are expected to be equally present in positive and negative samples, making the classification task more challenging. Interestingly, we observe that RNAProt shows the greatest drop performance when moving from the negative-1 to negative-2 setting. This may be explained by conservation scores (an auxiliary input of RNAProt) being higher for binding sites of the target RBP compared to random transcriptome sequences (negative-1), while being large similar to binding sites of other RBPs (negative-2).

Multi-task outperform binary methods in the negative-2 setting

Except MultiRBP [34], multi-task models outperform binary classification models, with average AUROC of 0.8774 and 0.8825 across datasets for DeepRiPe [40] and Multi-resBind [41], respectively. Strikingly, both DeepRiPe and Multi-resBind outperform Pysster, the best binary classification method, by a margin of 0.0277 for DeepRiPe and 0.00328 for Multi-resBind in AUROC performance on the negative-2 setting. The comparably lower performance of MultiRBP (average auROC of 0.7937) may be explained by the fact that this method was initially trained and optimized on *in vitro* RNAcompete data and trained for a fixed number of 78 epochs. Training with the same number of epochs on *in vivo* datasets with a varying number of experiments lead to a considerable degree of over-fitting, as unlike DeepRiPe and Multi-resBind, no early-stopping procedures were used. In addition, MultiRBP operates on a considerably smaller input size of 75nt, compared to 150nt in case of DeepRiPe and Multi-resBind, which may further impact performance. Given the higher performance of binary classification models in the negative-1 setting compared to the negative-2 setting, we speculated that a similar trend may be observed when adding negative-1 to the training of multi-task methods. To this end, we retrained DeepRiPe and Multi-resBind with an additional negative-1 label, by intersecting input tiles with negative-1 sample locations and retaining only those tiles which exclusively overlapped with negative-1 samples (Sections 3.2.5, DeepRiPe and 3.2.11, Multi-resBind). Indeed, Figure 2D and 2E show that addition of a universal negative label increases performance of both DeepRiPe and Multi-resBind by and average of 0.0329 and 0.0349, respectively. Note that even with addition of negative-1 labels, multi-task performances are not directly comparable to the binary classification negative-1 setting, as the later makes exclusive use of negative-1 samples for the negative class, which is not the case for multi-task models.

Sequence conservation and exon/intron information boosts performance

Given that RNAProt showed considerably higher performance than other binary models, we investigated whether this is due to the unique use of conservation scores (in terms of PhyloP and phastCons scores) and exon/intron annotations as auxiliary input. To this end, we removed all auxiliary inputs from RNAProt and re-trained it on sequence inputs only. Indeed, we observe a large drop in performance in the negative-1 (AUROC of 0.8857, drop of 0.056, Figure 3A) and negative-2 (AUROC of 8111, drop of 0.0457, Figure 3B) settings, confirming that RNAProt's auxiliary inputs improve performance significantly.

Input size matters

Given that Pysster performed best among binary classification methods, we investigated the impact of Pysster-specific model properties. We next performed a closer evaluation of the second most performative model, Pysster, which does not rely on any auxiliary inputs beyond RNA sequence. A distinctive feature of Pysster is its large input size of 400nt, while other methods such a PrismNet, iDeepS and deepRAM operate on a significantly smaller input size of 101nt. To test whether input size is the driver of Pysster's high performance, we re-trained Pysster across all datasets on an input size of 101nt. Indeed, reducing Pysster's input size to 101nt lead to a considerable drop of performance (Figure 3C and 3D), such that it now performs on-par with other deep learning binary classification methods such as PrismNet. This effect is maintained when training and evaluating models in the negative-2 setting. Here, we hypothesize that this could due to long-range effects that govern binding of proteins to RNA, such that models benefit from large input windows around potential binding sites. However, low resolution of called peaks across CLIP-seq experiments may also explain this effect, as this could lead to some binding sites not being contained in the model inputs, with an increase in input size alleviating this effect.

Effects of RNA structure as auxiliary input

Given that methods which utilize predicted or experimentally determined RNA structure as auxiliary input did not show a higher performance over sequence-only methods (Figure 2A and B), we investigated in more detail whether RNA structure lead to an increase in performance for individual RBPs. To this end, we compared the performance of structure-aware deep learning methods (iDeepS [30] and PrismNet [51]) with the mean performance of three sequence-only methods (Pysster [28], DeepCLIP [50] and DeepRAM [49]). Note that we used Pysster models trained on 101nt sequence inputs in order to remove any input-size related effects, as iDeepS, PrismNet and deepRAM were trained of sequences of size 101nt. Figure 3E and G depicts the difference in performance between iDeepS and PrismNet and sequence-only binary classification methods, respectively. While structure does not improve performance for a majority of RBPs, the figures show several outliers for which performance of sequence and structure based methods is elevated above sequence-only methods, including. Examples of RBPs that appear to benefit from structure information include EWSR1, PUS1 and CAPRIN1 for which higher performance is observed both for iDeepS and PrismNet. To evaluate whether similar structure-sensitive RBPs show an elevated performance for across both sequence+structure tools, we tested whether the top-10% of RBPs with the highest increase in performance in one tool are enriched in the top-10% of RBPs in the other. Among the 31 top-10% models, 13 were shared between

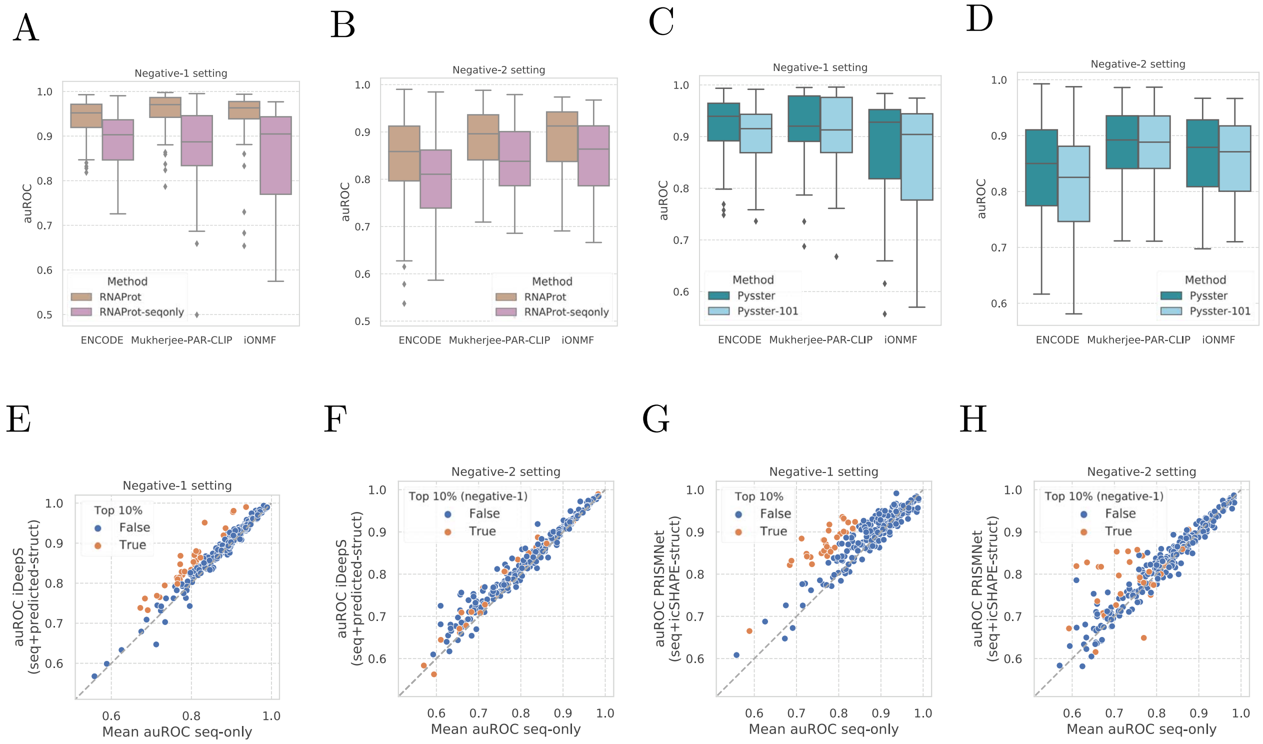


Figure 3. Exploration of methods' design choice. (**A** and **B**) Comparison of RNAProt vs 'RNAProt-seqonly' version where the additional sequence conservation scores and exon-intron annotations are not used, in the negative-1 setting (A) and negative-2 setting (B). (**A–D**) Comparison of Pysster vs 'Pysster-101' version where the sequence length of input is reduced from 400 to 101 nucleotides, in the negative-1 setting (A, C) and negative-2 setting (B, D). PRISMNet models are included in both for comparison. (**C–F**) Comparison of the average AUROC per RBP from models learning from sequence only (DeepCLIP, DeepRAM, Pysster-101) against AUROCs from iDeepS, a model learning from both sequence and predicted structure. Colored are the top 10% RBPs models showing a greater AUROC from models learning from both modalities under the negative-1 setting (C, E). The same models are colored in the negative-2 setting (D, F) for comparison. (**E–H**) Same set of plots, here comparing the average AUROC per RBP from models learning from sequence only against AUROCs from PRISMNet, a method learning from both sequence and *in vivo* measured structure. Colored are the top 10% RBPs models showing a greater AUROC from models learning from both modalities under the negative-1 setting (E, G). The same models are colored in the negative-2 setting (F, H) for comparison.

iDeepS and PrismNet in the negative-1 setting. A subsequent one-sided hypergeometric test showed that this enrichment is highly significant ($P = 5.9 \times 10^{-8}$). For the negative-2 setting, we observed a slightly smaller, albeit significant, overlap of 8/31 ($P = 1.6 \times 10^{-3}$). We next investigated whether similar structure-sensitive RBPs are recovered in both negative settings. As before, we compute the overlap of top-10% RBPs between the negative-1 and negative-2 setting and estimate the significance of this overlap via a one-sided hypergeometric test. For PrismNet, we observe an overlap of 13/30 ($P = 5.9 \times 10^{-8}$) RBPs, while for iDeepS we observe 6/31 ($P = 2.7 \times 10^{-2}$). We further investigated whether structure-sensitive RBPs are consistent across eCLIP datasets by selecting and comparing performance trends across a set of 73 intersecting RBPs between both ENCODE cell types (HepG2 and K562). For those RBPs, we computed the delta-AUROC (difference in performance between sequence and sequence+structure models) and evaluated whether the delta-auROC is consistent for RBPs across cell types. Supplementary Figure 1 shows the correlation delta-auROC scores for both negative sets and across iDeepS and PrismNet. A moderate to high correlation, ranging from Spearman's rho of 0.282 (negative-1, iDeepS) to 0.679 (negative-1, PrismNet) could be observed across the four settings, with PrismNet showing considerably higher correlation across cell types than iDeepS in the negative-1 setting, while showing a slightly lower correlation in the negative-2 setting. Interestingly, while PrismNet shows a strong drop in correlation when moving from the negative-1 to the negative-2 setting, iDeepS shows a marginal correlation increase. This confirms that structure-sensitive RBPs are consistent across eCLIP datasets of two cell types. The fact that predicted and experimentally measured RNA structure appears to be less informative for the negative-2 (compared to the negative-1) setting suggests that structural features primarily improve discrimination between protein-bound and unbound sites (negative-1), but not between sites bound by two or more different proteins (negative-2). Indeed, the majority of RBPs preferentially bind to single-stranded RNA [12], while Gosai *et al.* [79] further demonstrated anti-correlation between RBP binding and RNA structure *in vivo*. Thus, RNA structure may encode universal properties of RBP-binding sites, rather than RBP-specific information.

Method performance is correlated across CLIP-seq experiments

We next investigated how training and evaluation data affects predictive performance across methods. Figure 4A depicts the AUROC performance of each method across all CLIP-seq experiment as a function of the median performance of methods for the given experiment. One can observe that the performance per method across experiments correlates strongly with their median AUROC across methods. Notably, the variance in AUROC across methods is greater for experiments with overall lower performance as compared to high-performing experiments (bottom half versus top half - Levene statistics = 394.77; *P*-value < 1.139e-82). This effect is pronounced for the negative-2 setting (Figure 4B), which shows a strong linear correlation between

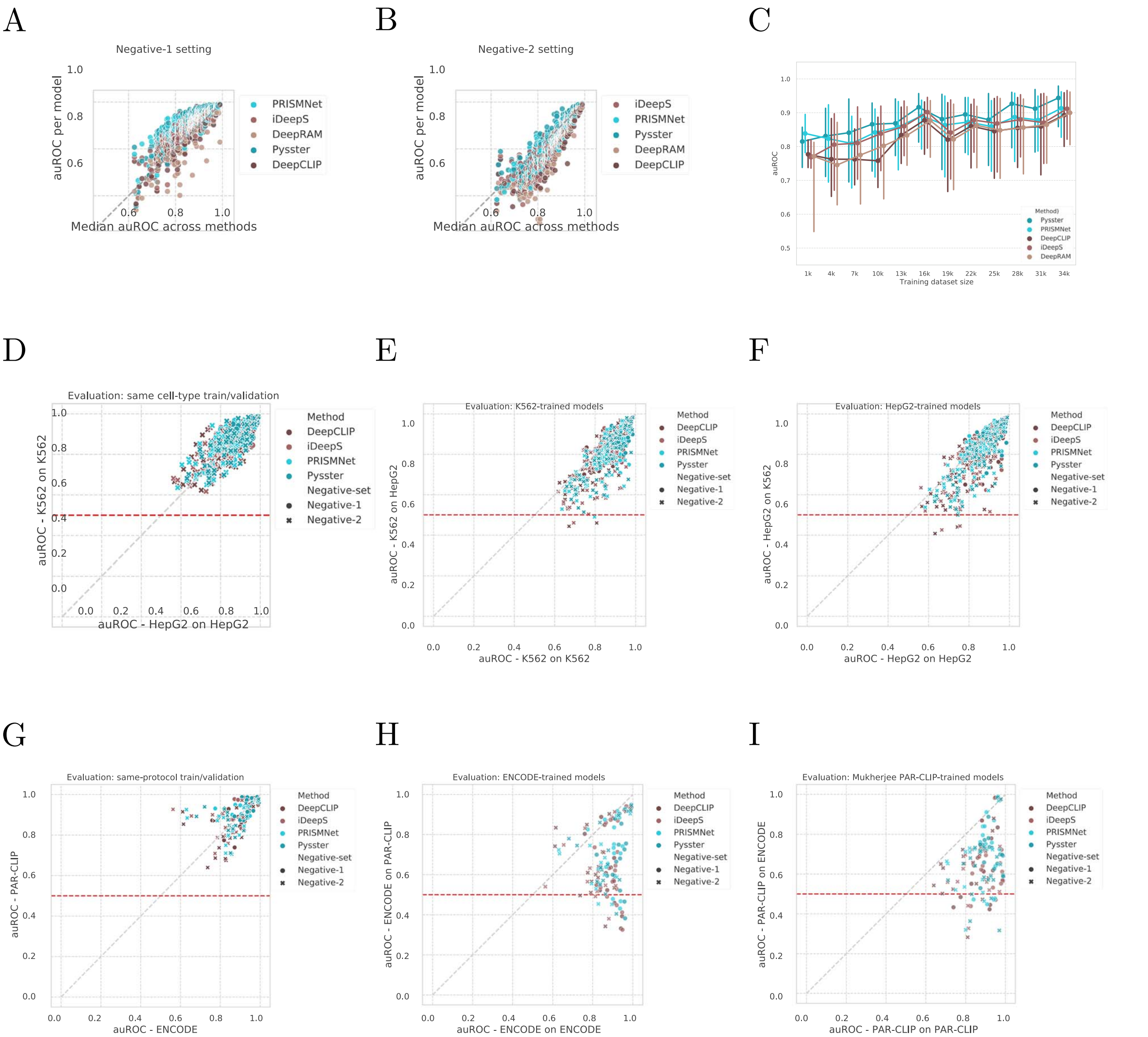


Figure 4. Influence of input modalities. (**A** and **B**) Stability of models' performance per RBP across single-label architectures. Each point is a model for an RBP and a method, plotted against the RBP's median AUROC across methods, in the negative-1 (A) and negative-2 (B) settings. (**C**) Correlation between model performance and dataset size, over the range of dataset sizes from ENCODE. Models are grouped per training dataset bin size (bin size = 2000). Dots represent the median AUROC of models per bin, for each method. Error bar: 25–75% interquartile. (**D**) Comparison of AUROCs for 73 RBPs evaluated in two different cell-types from the ENCODE dataset. Models are paired on the RBP names, while the AUROCs are computed on sequences derived from the same-cell type used for training. (**E** and **F**) Comparison of auROCs for 73 RBPs evaluated in two different cell-types from the ENCODE dataset, comparing the performance on same-cell-type evaluation (x-axis) against the performance from cross-cell-type evaluation (y-axis) for K562-trained models (E) and HepG2-trained models (**F**). Red line: random performance. (**G**) Comparison of auROCs for 17 RBPs matched between Mukherjee's PAR-CLIP and ENCODE eCLIP experiments. Models are paired on the RBP names, while the auROCs are computed on sequences derived from the same experimental-protocol used for training. (**H** and **I**) Comparison of AUROCs for 17 RBPs matched between Mukherjee's PAR-CLIP and ENCODE eCLIP experiments, comparing the performance on same-protocol evaluation (x-axis) against the performance from cross-protocol evaluation (y-axis) for ENCODE trained models (H) and PAR-CLIP trained models (I). Red line: random performance.

a method's performance and the median performance across methods for an experiment. This is likely due to different training set sizes across experiments, which vary greatly and have a significant effect on model performance (Figure 4C). As expected, we measured a significant positive correlation of training set size and model performance for both the ENCODE and the PAR-CLIP datasets (ENCODE: Spearman $r = 0.396$, $P < 1.829 \times 10^{-84}$; PAR-CLIP: $r = 0.196$, $P < 1.345 \times 10^{-5}$), while the iONMF dataset was excluded, as it has the same training set sizes across all experiments.

Models partially learn cell type specific binding

The ENCODE dataset consists of 223 eCLIP experiments across two cell lines, HepG2 (103) and K562 (120), with 73 RBPs being covered by both cell lines. Figure 4D compares the performance of DeepCLIP, iDeepS, PrismNet and Pysster models across RBPs covered by both ENCODE cell lines. Models did not perform better in one cell line over the other and this effect is consistent over negative-1 and negative-2 samples. We next turned to the question whether models trained on one cell type are applicable (i.e. retain high prediction performance) on another. This is crucial, as a key use case of computational methods for protein–RNA interaction prediction is the imputation of missing binding information on transcripts not present in the experimental condition, such as unexpressed transcripts. To this end, we selected RBPs covered by both HepG2 and K562 eCLIP experiments and performed cross-predictions, such that models trained on the HepG2 cell line were evaluated on hold-out data from the K562 cell line and vice versa. Figure 4E and F shows the cross-cell-line performance of model training on the HepG2 and K562 cell lines, respectively. We observed a performance drop of 4% and 7% for negative-1 and negative-2, respectively, indicating that machine learning models may learn cell-type-specific binding feature, which only partially generalize to other cell types (Supplementary Figure 3a). While in some cases, models fall to or even below the random baseline AUROC performance of 0.5 (red line), high performing models generally appear to yield high performance even in a cross-cell-line evaluation setting. This may suggest that RBPs with clearly defined binding motifs (i.e. RBPs with stable binding preferences) are easy to predict, even across cell types or that models trained on high-quality data tend to perform well both within and across cell-type prediction, or both.

Models learn strong protocol-specific biases

We next evaluated whether model performance is subject to the underlying CLIP-seq experimental protocol. To this end, we compared model performances for models trained on RBPs represented by both ENCODE eCLIP and PAR-CLIP experiments from the Mukherjee *et al*. [39] dataset. While performance differs between protocols for selected RBPs, there appears to be no general trend of better performance on data from one protocol over the other, as shown in Figure 4G. In analogy to our cross-cell-line evaluation, we next evaluated the extent to which models trained on data from one CLIP-seq protocol generalize to data from another protocol. Performance dropped significantly (average drop of 24% and 22% for negative-1 and negative-2, respectively; Supplementary Figure 3b) when evaluating trained model on data obtained from a different CLIP-seq protocol, as can be seen in Figure 4H and I for ENCODE and PAR-CLIP models, respectively. We note that besides protocol, ENCODE and Mukherjee *et al*. make use of different peak callers (CLIPper and PARalizer, respectively), which may impact the final set of binding sites significantly. Further analysis of the impact of peak callers are necessary in order to disentangle the effects of protocol and peak callers on performance drops observed here.

Limitations and future directions

While we sought to benchmark methods in a systematic and unbiased manner, some caveats exist. Several methods employ a hyperparameter search to obtain the optimal set of hyperparameters for a given dataset. As we evaluated methods across a large and diverse set of CLIP-seq experiments, including different protocols and cell lines, additional tuning of hyperparameters was not feasibly, as it would imply the training of tens of thousands of models. Nevertheless, it is important to note that the original method's hyperparameters may not perfectly translate to our benchmark data. Further, while most methods monitor the validation loss for the purpose of early stopping, some (GraphProt, MultiRBP and DeepRAM) instead suggest a default number of training epochs. If calibrated wrongly, this can lead to over-or under-fitting and thus reduce the methods performance. A key finding of this study is that input modalities appear to be more important for good model performance than the deep learning architecture. For instance, we do not find a notable difference between RNN and CNN architectures. Future studies may further probe the effects of model architecture in a more controlled setting, for instance by keeping the model parameters and input modalities constant.

The goal of this benchmark is to establish an evaluation framework for protein–RNA interaction prediction and to aid researchers in choosing the state-of-the-art method. To achieve this, methods are compared with respect to their classification performance, however, in practice researchers may be interest in tasks beyond the classification of individual sequences. For instance, methods may be utilized to score the impact of sequence variants on RBP-binding or to identify all binding sites across a transcript via a sliding-window approach. While likely correlated, maximizing performance on the RBP-binding classification task maybe not maximize performance for those tasks. As a future direction, we envision to expand the this benchmarking framework to probe methods performances on such auxiliary tasks. Several methods provide outputs beyond classification labels. For instance, GraphProt and DeepCLIP provide pseuo-nucleotide-resolution predictions as additional output modalities. On the other hand, PrismNet and RNAProt require icSHAPE and sequence conservation information as input modalities, which may not be available for all sequences. Thus, additional output and input modalities may influence the choice of method in practice, beyond classification performance.

CONCLUSION

In this study, we evaluate 11 *in vivo* protein–RNA interaction prediction methods across 313 CLIP-seq datasets with respect to their classification performance on a large cohort of CLIP-seq datasets. Our study revealed that among benchmarked methods, no particular deep learning architecture, such as CNN or RNN, represents a major advantage over others. However, our results showed that sequence conservation information and exon/intro annotation, as well as the size of the RNA input has a strong effect on model performance and that multi-task generally outperform single-task methods. We further explored two generation schemes for negative class samples and demonstrated that sampling negatives from unbound regions generally leads to higher performance, possibly due to incorporation of CLIP-seq biases as discriminative features. We demonstrated that predicted and *in vivo* secondary structure might improve model performance for some RBPs, while this effect is subject to the chosen negative samples and is diminished in case negatives are sampled from binding sites of other RBPs. Cross-evaluation results showed that models partially learned cell-type specific RBP-binding, while prediction across protocols leads to a strong decrease in performance, which may be attributed to protocol-specific biases or the use of different peak callers. We believe that this study will guide the development of future methods in the field of computational modeling of protein–RNA interaction by serving as a reference for method design in regards to architecture, input modalities and generation of negative controls.

Key Points

- A variety of deep learning methods have been developed in the past years to learn and predict protein–RNA interaction

- We designed a benchmark framework that unifies pre-processing, control sampling and train/test splitting of CLIP-seq datasets to enable unbiased comparison of 11 methods
- We show that multi-task models dominate single-task models and demonstrate that sequence conservation scores and exon/intron annotations boost performance considerably
- Cross-evaluations and comparison of negative-sampling schemes suggest that models may learn varying levels of protocol- and cell-type specific biases, leading to decreased performance during cross-prediction

SUPPLEMENTARY DATA

Supplementary data are available online at http://bib.oxfordjournals.org/.

FUNDING

This work was supported by the Helmholtz Association under the joint research school 'Munich School for Data Science (MUDS)' to M.H., G.C., P.S. and A.M. and the 'Deutsche Forschungsgemeinschaft' (SFB/TR501 84 TP C01) to A.M. and L.M.

DATA AVAILABILITY

All data processed in this study was obtained exclusively from public sources (ENCODE [7], Mukherjee *et al.* [39] and Stražar *et al.* [61]).

CODE AVAILABILITY

Code for reproducing the results of this study is available at https://github.com/mhorlacher/Benchmark-RBP.

REFERENCES

1. Gerstberger S, Hafner M, Tuschl T. A census of human RNA-binding proteins. **15**(12):829–45.
2. Van Nostrand EL, Pratt GA, Shishkin AA, *et al.* Robust transcriptome-wide discovery of RNA-binding protein binding sites with enhanced clip (eclip). *Nat Methods* 2016; **13**(6): 508–14.
3. Gebauer F, Schwarzl T, Valcárcel J, Hentze MW. RNA-binding proteins in human genetic disease. **22**(3):185–98.
4. Lee FCY, Ule J. Advances in clip technologies for studies of protein-RNA interactions. *Mol Cell* 2018; **69**(3):354–69.
5. Danan C, Manickavel S, Hafner M. Par-clip: a method for transcriptome-wide identification of RNA binding protein interaction sites. *Methods Mol Biol* 2016;(1358):153–73.
6. König J, Zarnack K, Rot G, *et al.* Iclip reveals the function of hnrnp particles in splicing at individual nucleotide resolution. *Nat Struct Mol Biol* 2010; **17**(7):909–15.
7. Van Nostrand EL, Freese P, Pratt GA, *et al.* A large-scale binding and functional map of human RNA-binding proteins. *Nature* 2020; **583**(7818):711–9.
8. Yan J, Zhu M. A review about RNA–protein-binding sites prediction based on deep learning. *IEEE Access* 2020; **8**:150929–44.
9. Pan X, Yang Y, Xia C-Q, *et al.* Recent methodology progress of deep learning for RNA–protein interaction prediction. *Wiley interdisciplinary reviews. RNA* 2019; **10**(6):e1544.
10. Wei J, Chen S, Zong L, *et al.* Protein–RNA interaction prediction with deep learning: structure matters. *Brief Bioinform* 2022; **23**(1): bbab540.
11. Eraslan G, Avsec Z, Gagneur J, Theis FJ. Deep learning: new computational modelling techniques for genomics. *Nat Rev Genet* 2019; **20**(7):389–403.
12. Jolma A, Zhang J, Mondragón E, *et al.* Binding specificities of human RNA-binding proteins toward structured and linear RNA sequences. *Genome Res* 2020; **30**(7):962–73.
13. Wheeler EC, Van Nostrand EL, Yeo GW. Advances and challenges in the detection of transcriptome-wide protein–RNA interactions. *Wiley interdisciplinary reviews. RNA* 2018; **9**(1): e1436.
14. Orenstein Y, Hosur R, Simmons S, *et al.* Sequence biases in clip experimental data are incorporated in protein RNA binding models. *bioRxiv* 2016;075259.
15. Hafner M, Landthaler M, Burger L, *et al.* Transcriptome-wide identification of RNA-binding protein and microRNA target sites by par-clip. *Cell* 2010; **141**(1):129–41.
16. Bailey TL, Elkan C. Fitting a mixture model by expectation maximization to discover motifs in biopolymers. *Proc Int Conf Intell Syst Mol Biol*. 1994;**2**:28–36.
17. Foat BC, Morozov AV, Bussemaker HJ. Statistical mechanical modeling of genome-wide transcription factor occupancy data by matrixreduce. *Bioinformatics* 2006; **22**(14): e141–9.
18. Hiller M, Pudimat R, Busch A, Backofen R. Using RNA secondary structures to guide sequence motif finding towards single-stranded regions. *Nucleic Acids Res* 2006; **34**(17): e117–7.
19. Kazan H, Ray D, Chan ET, *et al.* Rnacontext: a new method for learning the sequence and structure binding preferences of RNA-binding proteins. *PLoS Comput Biol* 2010; **6**(7): e1000832.
20. Maticzka D, Lange SJ, Costa F, Backofen R. Graphprot: modeling binding preferences of RNA-binding proteins. *Genome Biol* 2014; **15**(1):1–18.
21. Dominguez D, Freese P, Alexis MS, *et al.* Sequence, structure, and context preferences of human RNA binding proteins. *Mol Cell* 2018; **70**(5):854–67.
22. Essig K, Kronbeck N, Guimaraes JC, *et al.* Roquin targets mrnas in a 3′-utr-specific manner by different modes of regulation. *Nat Commun* 2018; **9**(1): 3810.
23. Hofacker IL. Vienna RNA secondary structure server. *Nucleic Acids Res* 2003; **31**(13):3429–31.
24. Steffen P, Voß B, Rehmsmeier M, *et al.* Rnashapes: an integrated RNA analysis package based on abstract shapes. *Bioinformatics* 2006; **22**(4):500–3.
25. Deng L, Liu Y, Shi Y, *et al.* Deep neural networks for inferring binding sites of RNA-binding proteins by using distributed representations of RNA primary sequence and secondary structure. *BMC Genomics* 2020; **21**(13):1–10.
26. Shen Z, Deng S-P, Huang D-S. Capsule network for predicting RNA-protein binding preferences using hybrid feature. *IEEE/ACM Trans Comput Biol Bioinform* 2019; **17**(5):1483–92.
27. Gandhi S, Lee LJ, Delong A, *et al.* Cdeepbind: a context sensitive deep learning model of RNA-protein binding. *bioRxiv* 2018;345140.
28. Budach S, Marsico A. Pysster: classification of biological sequences by learning sequence and structure motifs with convolutional neural networks. *Bioinformatics* 2018; **34**(17): 3035–7.
29. Zhang S, Zhou J, Hailin H, *et al.* A deep learning framework for modeling structural features of RNA-binding protein targets. *Nucleic Acids Res* 2016; **44**(4): e32–2.

30. Pan X, Rijnbeek P, Yan J, Shen H-B. Prediction of RNA-protein sequence and structure binding preferences using deep convolutional and recurrent neural networks. *BMC Genomics* 2018; **19**(1):1–11.
31. Petrov AI, Zirbel CL, Leontis NB. Automated classification of RNA 3D motifs and the RNA 3D motif atlas. *RNA* 2013; **19**(10): 1327–40.
32. Li Z, Zhu J, Xiaojiang X, Yao Y. Rdense: a protein-RNA binding prediction model based on bidirectional recurrent neural network and densely connected convolutional networks. *IEEE Access* 2019; **8**:14588–605.
33. Ben-Bassat I, Chor B, Orenstein Y. A deep neural network approach for learning intrinsic protein-RNA binding preferences. *Bioinformatics* 2018; **34**(17): i638–46.
34. Karin J, Michel H, and Orenstein Y. Multirbp: multi-task neural network for protein-RNA binding prediction. In: *Proceedings of the 12th ACM Conference on Bioinformatics, Computational Biology, and Health Informatics*, BCB '21, New York, NY, USA, 2021. Association for Computing Machinery.
35. Pan X, Shen H-B. RNA-protein binding motifs mining with a new hybrid deep learning based cross-domain knowledge integration approach. *BMC Bioinformatics* 2017; **18**(1):1–14.
36. Yan Z, Hamilton WL, Blanchette M. Graph neural representational learning of RNA secondary structures for predicting RNA-protein interactions. *Bioinformatics* 2020; **36**(Supplement_1): i276–84.
37. Chung T, Kim D. Prediction of binding property of RNA-binding proteins using multi-sized filters and multi-modal deep convolutional neural network. *PloS One* 2019; **14**(4):e0216257.
38. Flynn RA, Zhang QC, Spitale RC, *et al.* Transcriptome-wide interrogation of RNA secondary structure in living cells with icSHAPE. *Nat Protoc* 2016; **11**(2):273–90.
39. Mukherjee N, Wessels H-H, Lebedeva S, *et al.* Deciphering human ribonucleoprotein regulatory networks. *Nucleic Acids Res* 2019; **47**(2):570–81.
40. Ghanbari M, Ohler U. Deep neural networks for interpreting RNA-binding protein target preferences. *Genome Res* 2020; **30**(2): 214–26.
41. Zhao S, Hamada M. Multi-resbind: a residual network-based multi-label classifier for in vivo RNA binding prediction and preference visualization. *BMC Bioinformatics* 2021; **22**(1):1–15.
42. Uhl M, Tran VD, Heyl F, Backofen R. Rnaprot: an efficient and feature-rich RNA binding protein binding site predictor. *GigaScience* 2021; **10**(8): giab054.
43. Avsec ž, Barekatain M, Cheng J, Gagneur J. Modeling positional effects of regulatory sequences with spline transformations increases prediction accuracy of deep neural networks. *Bioinformatics* 2018; **34**(8):1261–9.
44. Ray D, Kazan H, Cook KB, *et al.* A compendium of RNA-binding motifs for decoding gene regulation. *Nature* 2013; **499**(7457): 172–7.
45. Alipanahi B, Delong A, Weirauch MT, Frey BJ. Predicting the sequence specificities of DNA-and RNA-binding proteins by deep learning. *Nat Biotechnol* 2015; **33**(8):831–8.
46. Pan X, Yan J. Attention based convolutional neural network for predicting rna-protein binding sites. arXiv preprint arXiv:1712.02270. 2017.
47. Pan X, Shen H-B. Predicting RNA–protein binding sites and motifs through combining local and global deep convolutional neural networks. *Bioinformatics* 2018; **34**(20):3427–36.
48. Pan X, Fan Y-X, Jia J, Shen H-B. Identifying RNA-binding proteins using multi-label deep learning. *Sci China Inform Sci* 2019; **62**(1): 1–3.
49. Trabelsi A, Chaabane M, Ben-Hur A. Comprehensive evaluation of deep learning architectures for prediction of DNA/RNA sequence binding specificities. *Bioinformatics* 2019; **35**(14): i269–77.
50. Grønning AGB, Doktor TK, Larsen SJ, *et al.* Deepclip: predicting the effect of mutations on protein–RNA binding with deep learning. *Nucleic Acids Res* 2020; **48**(13):7099–118.
51. Sun L, Kui X, Huang W, *et al.* Predicting dynamic cellular protein–RNA interactions by deep learning using in vivo RNA structures. *Cell Res* 2021; **31**(5):495–516.
52. Hu J, Shen L, and Sun G. Squeeze-and-excitation networks. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7132–41, 2018.
53. Koo PK, Majdandzic A, Ploenzke M, *et al.* Global importance analysis: an interpretability method to quantify importance of genomic features in deep neural networks. *PLoS Comput Biol* 2021; **17**(5): e1008925.
54. Sharma NK, Gupta S, Kumar A, *et al.* Rbpspot: learning on appropriate contextual information for rbp binding sites discovery. *Iscience* 2021; **24**(12): 103381.
55. Pan X, Shen H-B. Learning distributed representations of RNA sequences and its application for predicting RNA-protein binding sites with a convolutional neural network. *Neurocomputing* 2018; **305**:51–8.
56. Mikolov T, Chen K, Corrado G, Dean J. Efficient estimation of word representations in vector space. arXiv preprint. arXiv:1301.3781. 2013.
57. Devlin J, Chang M-W, Lee K, Toutanova K. Bert: pre-training of deep bidirectional transformers for language understanding. arXiv preprint. arXiv:1810.04805. 2018.
58. Lin Z, Akin H, Rao R, *et al.* Evolutionary-scale prediction of atomic level protein structure with a language model. *bioRxiv* 2022; 2022–07.
59. Yamada K, Hamada M. Prediction of RNA–protein interactions using a nucleotide language model. *Bioinformatics Adv* 2022; **2**(1): vbac023.
60. Ji Y, Zhou Z, Liu H, Davuluri RV. Dnabert: pre-trained bidirectional encoder representations from transformers model for DNA-language in genome. *Bioinformatics* 2021; **37**(15): 2112–20.
61. Stražar M, žitnik M, Zupan B, *et al.* Orthogonal matrix factorization enables integrative analysis of multiple RNA binding proteins. *Bioinformatics* 2016; **32**(10):1527–35.
62. Tahir M, Tayara H, Hayat M, Kil To Chong. Kdeepbind: prediction of rna-proteins binding sites using convolution neural network and k-gram features. *Chemom Intel Lab Syst* 2021; **208**: 104217.
63. Zhang S-W, Wang Y, Zhang X-X, Wang J-Q. Prediction of the rbp binding sites on lncrnas using the high-order nucleotide encoding convolutional neural network. *Anal Biochem* 2019; **583**:113364.
64. Zhihua D, Xiao X, Uversky VN. Deepa-rbpbs: a hybrid convolution and recurrent neural network combined with attention mechanism for predicting rbp binding site. *J Biomolecular Struct Dynamics* 2022; **40**(9):4250–8.
65. Shen Z, Deng S-P, Huang D-S. RNA-protein binding sites prediction via multi scale convolutional gated recurrent unit networks. *IEEE/ACM Trans Comput Biol Bioinform* 2019; **17**(5):1741–50.
66. Licatalosi DD, Mele A, Fak JJ, *et al.* Hits-clip yields genome-wide insights into brain alternative rna processing. *Nature* 2008; **456**(7221):464–9.
67. Curk T, Rot G, Gorup C, *et al.* Icount: protein-RNA interaction iclip data analysis. *Prep* 2019.

68. Anders G, Mackowiak SD, Jens M, *et al.* Dorina: a database of RNA interactions in post-transcriptional regulation. *Nucleic Acids Res* 2012; **40**(D1): D180–6.
69. Corcoran DL, Georgiev S, Mukherjee N, *et al.* Paralyzer: definition of RNA binding sites from par-clip short-read sequence data. *Genome Biol* 2011; **12**(8):1–16.
70. Frankish A, Diekhans M, Jungreis I, *et al.* Gencode 2021. *Nucleic Acids Res* 2021; **49**(D1): D916–23.
71. Costa F, De Grave K. Fast neighborhood subgraph pairwise distance kernel. In *Proceedings of the 26th International Conference on Machine Learning*, 2010, (pp. 255–262).
72. Budach S. Explainable deep learning models for biological sequence classification (Doctoral dissertation), 2021.
73. Ray D, Kazan H, Chan ET, *et al.* Rapid and systematic analysis of the RNA recognition specificities of RNA-binding proteins. *Nat Biotechnol* 2009; **27**(7):667–70.
74. Yufeng S, Luo Y, Zhao X, *et al.* Integrating thermodynamic and sequence contexts improves protein–RNA binding prediction. *PLoS Comput Biol* 2019; **15**(9): e1007283.
75. Pollard K S, Hubisz M J, Rosenbloom K R, and Siepel A. Detection of nonneutral substitution rates on mammalian phylogenies. *Genome Res*, **20**(1):110–21, January 2010.
76. Siepel A, Bejerano G, Pedersen J S, Hinrichs A S, Hou M, Rosenbloom K, Clawson H, Spieth J, Hillier L W, Richards S, Weinstock G M, Wilson R K, Gibbs R A, James Kent W, Miller W, and Haussler D. Evolutionarily conserved elements in vertebrate, insect, worm, and yeast genomes. *Genome Res*, **15**(8):1034–50, August 2005.
77. Lovci MT, Ghanem D, Marr H, *et al.* Rbfox proteins regulate alternative mRNA splicing through evolutionarily conserved RNA bridges. *Nat Struct Mol Biol* 2013; **20**(12):1434–42.
78. Hafner M, Katsantoni M, Köster T, *et al.* Clip and complementary methods. *Nat Rev Methods Primers* 2021; **1**(1):1–23.
79. Gosai SJ, Foley SW, Wang D, *et al.* Global analysis of the RNA-protein interaction and RNA secondary structure landscapes of the arabidopsis nucleus. *Mol Cell* 2015; **57**(2):376–88.
80. Zhou J, Park C Y, Theesfeld C L, Wong A K, Yuan Y, Scheckel C, Fak J J, Funk J, Yao K, Tajima Y, Packer A, Darnell R B, and Troyanskaya O G. Whole-genome deep learning analysis identifies contribution of noncoding mutations to autism risk. *Nat Genet*, **51**(6):973–80, June 2019.
81. Yang H, Deng Z, Pan X, *et al.* RNA-binding protein recognition based on multi-view deep feature and multi-label learning. *Brief Bioinform* 2021; **22**(3): bbaa174.
82. Liu Y, Li R, Luo J, Zhang Z. Inferring RNA-binding protein target preferences using adversarial domain adaptation. *PLoS Comput Biol* 2022; **18**(2): e1009863.
83. Zhang J, Liu B, Wang Z, *et al.* Deeppn: a deep parallel neural network based on convolutional neural network and graph convolutional network for predicting RNA-protein binding sites. 2022.
84. Dassi E, Re A, Leo S, Tebaldi T, Pasini L, Peroni D, and Quattrone A. Aura 2. *Translation*, **2**(1), J2014.
85. Corrado G, Tebaldi T, Costa F, *et al.* RNAcommender: genome-wide recommendation of RNA–protein interactions. *Bioinformatics* 2016; **32**(23):3627–34.

9 RBPNET: SINGLE-NUCLEOTIDE PREDICTION OF RBP-BINDING

Towards In-Silico CLIP-seq: Predicting Protein-RNA Interaction via Sequence-to-Signal Learning

Authors: Marc Horlacher, Nils Wagner, Lambert Moyon, Klara Kuret, Nicolas Goedert, Marco Salvatore, Jernej Ule, Julien Gagneur, Ole Winther, Annalisa Marsico

Journal: Genome Biology, 2023 [3]

Summary: Experimental protocols, such as CLIP, allow for accurate transcriptome-wide profiling of in vivo protein-RNA interactions. However, as only a fraction of transcripts are expressed in the experimental cell type at any given time, it draws an incomplete picture of an RBP's binding landscape, creating the need for computational imputation of missing binding information. Existing classification-based methods predict binding with low resolution and depend on prior labeling of transcriptome regions for training. On the path towards computational imputation of protein-RNA interaction from RNA sequence at a spatial resolution that matches CLIP data, a natural, but so far unexplored avenue is to model the CLIP signal directly. This publication presents RBPNet, a novel deep learning method which, given an RNA sequence, predicts the CLIP signal distribution at single-nucleotide resolution. RBPNet directly operates on the raw CLIP signal, eliminating the need for peak calling for training set construction. To prevent learning of unspecific background and bias signal, RBPNet optionally models the signal of a paired control experiment alongside the CLIP signal. Specifically, RBPNet attempts to explain the observed CLIP signal as an additive mixture of two signal components - the control, which is explicitly learned from the control experiment, and an unobserved target component, which represents the protein-specific signal. The total CLIP signal is given by a mixture of the protein-specific and control components, thus mitigating technical biases which may otherwise hinder downstream analysis. The study demonstrates that RBPNet is broadly applicable across CLIP protocols, including eCLIP, iCLIP and m6A Individual-nucleotide CLIP (miCLIP). Evaluation on over 100 eCLIP experiments showed that RBPNet achieves high generalization performance, outperforming state-of-the-art classifiers with respect to recovering experimentally identified protein-RNA crosslink sites by a large margin. Crucially, RBPNet allows for prediction on variable-length RNA sequences, up to lengths of hundreds of thousands of nucleotides and thus enables inference on whole transcripts in a single model forward pass. By attributing RBPNet predictions to each input nucleotide via Integrated Gradients importance scoring, predictive RNA sub-sequences are extracted from binding regions and compiled to consensus motifs, which are shown to be strongly correlated with motifs found via *in vitro* assays. Finally, using in silico mutagenesis, RBPNet scores the impact of single-nucleotide variants on RBP-binding by comparing the predicted binding profiles for references and alternative alleles. RBPNet is the first method to directly model the raw CLIP signal at nucleotide-resolution, thus improving both computational inference of protein-RNA interaction as well as interpretation of predictions over the state-of-the-art.

Availability: Code to replicate the models in this study can be found at https://github.com/mhorlacher/RBPNet. A web-server is available at https://biolib.com/mhorlacher/RBPNet to perform RBPNet predictions on user-provided sequences for 103 RBPs.

Contributions: Study design and conceptualization; Processing of eCLIP data from the ENCODE database, as well as iCLIP and miCLIP data; Design and software implementation of the RBPNet model, including training and evaluation scripts as well as software for variant impact prediction with RBPNet; Model training and monitoring of training runs on HPC clusters; Analysis and aggregation of results and figure design, in particular the design of main figures 1-5 and 7; Preparation of the manuscript text and formatting of the submission.

Horlacher *et al. Genome Biology* (2023) 24:180
https://doi.org/10.1186/s13059-023-03015-7

Genome Biology

METHOD

Open Access

Towards in silico CLIP-seq: predicting protein-RNA interaction via sequence-to-signal learning

Marc Horlacher[1,2,3,4*], Nils Wagner[3,4], Lambert Moyon[1], Klara Kuret[5,6,7], Nicolas Goedert[1], Marco Salvatore[2], Jernej Ule[5,6], Julien Gagneur[1,3,4], Ole Winther[2*] and Annalisa Marsico[1,4*]

*Correspondence: marc.horlacher@helmholtz-muenchen.de; ole.winther@bio.ku.dk; annalisa.marsico@helmholtz-muenchen.de

[1] Computational Health Center, Helmholtz Center Munich, Munich, Germany
[2] Department of Biology, University of Copenhagen, Copenhagen, Denmark
[3] Department of Informatics, Technical University of Munich, Garching, Germany
[4] Helmholtz Association - Munich School for Data Science (MUDS), Munich, Germany
[5] National Institute of Chemistry, Ljubljana, Slovenia
[6] The Francis Crick Institute, London, UK
[7] Jozef Stefan International Postgraduate School, Jamova cesta 39, 1000 Ljubljana, Slovenia

Abstract

We present RBPNet, a novel deep learning method, which predicts CLIP-seq crosslink count distribution from RNA sequence at single-nucleotide resolution. By training on up to a million regions, RBPNet achieves high generalization on eCLIP, iCLIP and miCLIP assays, outperforming state-of-the-art classifiers. RBPNet performs bias correction by modeling the raw signal as a mixture of the protein-specific and background signal. Through model interrogation via Integrated Gradients, RBPNet identifies predictive sub-sequences that correspond to known and novel binding motifs and enables variant-impact scoring via in silico mutagenesis. Together, RBPNet improves imputation of protein-RNA interactions, as well as mechanistic interpretation of predictions.

Keywords: CLIP-seq, Deep learning, Computational biology, Protein-RNA interaction

Background

RNA-binding proteins (RBPs) are a family of proteins that bind to coding and non-coding transcripts, usually through recognition of short sequence or structural features commonly known as motifs [10]. To date, over 2, 000 proteins have been experimentally identified as RNA-binding, rendering it one of the largest cellular protein groups [17]. RBPs are involved in every aspect of post-transcriptional regulation, including modification, stabilization, localization, splicing, and translation of RNAs [25]. Misregulation of RBPs, as well as mutations in their amino acid sequence or the sequence of their RNA targets, has been associated with an abundance of human diseases, including neurological and psychiatric disorders [49]. Therefore, uncovering binding preferences and RNA targets of RBPs is crucial for understanding the role of RBPs in post-transcriptional regulatory pathways and for quantifying the impact of their dis-regulation in context of human disease. Nowadays, the most common protein-centric experimental approach to profile RNA-binding in vivo is via individual-nucleotide resolution Cross-Linking and

Immunoprecipitation followed by sequencing (CLIP-seq) [29, 61] and its derivatives, which enables transcriptome-wide detection of protein-RNA interactions for a protein of interest. Variants of CLIP-seq, including *individual-nucleotide* and *enhanced* CLIP (iCLIP and eCLIP, respectively), further allow for detection of protein-RNA crosslinking events at single-nucleotide resolution. Here, in brief, cells are radiated with UV light, forming covalent cross-links between RNA molecules and bound proteins. Protein/RNA complexes are then purified with protein-specific antibodies and the bound proteins are digested. Subsequently, the bound RNA molecules are reverse-transcribed to cDNA, followed by high-throughput sequencing. Reverse-transcription often truncates at the cross-linked site due to a small peptide remaining bound to the site after protein digestion. After alignment of reads to the reference genome, this leads to an accumulation of read-starts at one nucleotide down-stream of the cross-linked site. The resulting nucleotide-wise count signal can then be used to identify binding sites of the RBP of interest as RNA regions where the signal is significantly higher than expected, given some background model [57]. CLIP-seq data is commonly post-processed with peak callers, which identify a number of regions of enriched signal (also referred to as *binding sites*), usually in the order of thousands. As peak calling is subject to unspecific cross-linking events in the underlying data, inferring target-specific signal from CLIP data is crucial. Multiple studies have identified a number of CLIP-associated biases, including background signal from abundant RNAs that are not properly washed during library preparation, library contamination with bound RNAs of other RBPs and strong UV crosslinking bias towards single-stranded uridine-rich motifs [22]. CLIP-seq experiments are often paired with a control, to account for unspecific background signal. For instance, the eCLIP protocol is designed to generate a size-matched input (SMInput) control by omitting the IP step such that the resulting library represents RNA fragments crosslinked to a mixture of background proteins with a similar molecular weight as the target RBP. Therefore, eCLIP SMInput data is a powerful resource to correct computational methods for experimental bias and reduce the number of detected unspecific binding events for an RBP of interest.

Analysis of experimentally identified RNA-binding sites can give insight into both the functional role of the RBP as well as the RNA sequence and structure feature by which it identifies (and binds to) its RNA targets. In addition, knowledge of the binding preference of an RBP enables imputation of protein-RNA interaction on RNAs not present at the time of the experiment. While traditional methods for de novo motif finding [3] or more sophisticated generative models [24] analyze identified RNA-binding sites directly, for instance by aggregating enriched sub-sequences into motif position-weight matrices (PWMs), more recent model-based approaches [32, 44] attempt to model RBP-binding as a function of RNA sequence for a given protein of interest. That is, model-based approaches attempt to find a function of the form $f : RNA \rightarrow \sigma(CLIP)$, where σ is some post-processing function on the raw experimental CLIP, for instance a peak caller. Model-based analysis of RBP binding has several key advantages over traditional model-free approaches. First, the availability of a model allows for imputation of missing binding site information. As only a fraction of transcripts may be expressed in the experimental cell type at a given time, CLIP-seq experiments generally draw an incomplete picture of an RBP's binding landscape. Here, predictions from RBP-binding models may aid researchers in generalizing their analysis to unobserved transcripts by providing candidate sites of protein-RNA interaction. The ability to impute missing binding information on arbitrary RNA sequences is especially relevant for the analysis of RBP-binding to foreign sequences, such as foreign RNAs derived from viruses [27]. Second, the model *f* represents a simplified abstraction of the in vivo biology, that is, the recognition of a binding site in a target RNA sequence by the protein of interest. Besides identifying drivers of RNA target recognition (e.g., binding motifs), the ability to study *f* enables "what-if" analysis, allowing one to explore how changes in the RNA input sequence affect RBP binding. For instance, researchers may investigate the impact of single-nucleotide variants (SNVs) on RBP binding in silico.

Deep learning enabled ground-breaking performance on tasks across a broad domain of research, including the computational modeling of protein-RNA interaction [1, 6, 18, 48]. Current state-of-the-art RBP-binding predictions models are generally classification-based, that is, given an input RNA sequence, the model is tasked with predicting whether the sequence is *bound* or *unbound* by the protein of interest. While classification-based models represent a significant improvement over traditional methods, they have several limitations. First, training and evaluation of classification-based models requires prior annotation of sequences with high-quality binary bound/unbound labels, generally through the use of peaks callers, making the model heavily dependent on upstream preprocessing steps. Therefore, performance of classification-based models is highly sensitive towards the availability of unbiased bona fide binding sites. Second, predictions of classification-based models generally have low resolution. While methods commonly take as input RNA sequences of 100s of nucleotides in length, the predicted label is assigned to the entire input region. This create ambiguity with regards to the exact location of the protein-RNA interaction site, which usually spans only a few nucleotides. Third, binary labels (*bound* and *unbound*) modeled by classification-based methods represent a strong simplification of the information yielded by CLIP-seq experiment. Compression of the CLIP-seq signal footprint within transcript region to a binary value may lead to loss of essential information for understanding the nuances of protein-RNA interaction.

Recently, a new class of models has emerged, which directly predict experimental signal from genomic sequences [2, 34]. For instance, BPNet [2] trains a dilated convolutional neural network which models transcription factor (TF) binding by predicting ChIP-nexus signal from DNA sequences at base resolution. However, there is a lack of sequence-to-signal models for prediction tasks on RNA sequences, including the task of modeling protein-RNA interaction via CLIP-seq signal prediction. In context of CLIP-seq, the presence of technical biases and cell-type specific RNA abundance pose a challenge with respect to the identification of sequence-mediated binding mechanisms at single-nucleotide resolution. Therefore, models need to account for the fact that the observed signal may partially be observed due to technical biases, rather than protein-RNA interaction of target RBP. This effect may further depend on the RNA sequence context, as is the case for nucleotide-specific crosslinking biases or sample contamination with other RBPs which themselves have certain sequence-depended binding preferences.

To fill this gap we developed RBPNet, a sequence-to-signal dilated convolutional neural network, which learns a direct mapping of RNA sequences to crosslink count signal extracted from CLIP-seq experiments. Given a variable-length RNA input sequence, RBPNet predicts the *distribution* of crosslink counts at single-nucleotide resolution along the input sequence, thereby enabling learning the binding profile of an RBP on a transcripts of arbitrary length. Existing sequence-based models of CLIP data are binary classifiers which necessitate important preprocessing steps, thus loosing the high resolution and the quantitative nature of the data In contrast to current state-of-the-art classifiers, RBPNet does not require a peak caller to produce transcript regions as candidate sites for model training. Instead, it uses a lenient cutoff-based approach to train on all regions that show an enrichment of crosslink counts, thereby making maximal use of signal generated by the underlying experiment. By additionally modeling the background signal of a paired control experiment, RBPNet implicitly learns the bias and protein-specific crosslinking components of the total signal, allowing one to disentangle genuine signal from noise. By training on hundreds of thousands of regions per RBP CLIP experiment, RBPNet reaches high accuracy in predicting RBP-binding signal shape on held-out test data across eCLIP, iCLIP, and miCLIP experiments. Furthermore, it allows for direct inference of predictions across variable-length sequences and whole transcripts, while outperforming state-of-the-art RBP-binding classification models with respect to the identification of crosslink sites derived from PureCLIP [38], a single-nucleotide peak caller. By performing model interpretation with Integrated Gradients, we demonstrate the capability of RBPNet to accurately identify the sequence patterns driving RBP-RNA interactions and enable binding motif discovery. Lastly, we show the high potential of RBPNet in scoring the impact of single-nucleotide genetic variants on RBP binding, and thereby enable prioritization of functional variants.

Results

RBPNet predicts crosslink count distribution from RNA sequence

RBPNet is a deep convolutional sequence-to-signal neural network for modeling protein-RNA interaction profiles, which takes as input a RNA sequence and outputs a probability vector of the same length, describing a discrete distribution of counts within that sequence. In the context of eCLIP, iCLIP, or other individual nucleotide CLIP technologies, RBPNet predicts the distribution of cDNA truncation events, as a result of protein-RNA crosslinking and hereafter also referred to as "crosslink counts," along an input RNA sequence. In this study, RBPNet was trained and evaluated on a large cohort of eCLIP, iCLIP, and miCLIP datasets. An outline of the study is shown in Fig. 1a, which gives a schematic overview of training data generation, model training and interrogation for investigating sequence determinants of RBP binding, including binding motif extraction.

The RBPNet model architecture consists of two major parts the model body, comprised of the input layer followed by several convolutional blocks with residual connections, and the model head, which performs the final mapping of the input sequence representation, derived from the body model output, to a probability vector (Fig. 1b). Importantly, while RBPNet is trained on fixed-length inputs, its purely convolutional architecture enables prediction on RNA sequences of arbitrary length. During training, the predicted probability vector is used to parameterize a multinomial distribution of crosslink counts and, given the position-wise observed counts in the input sequence

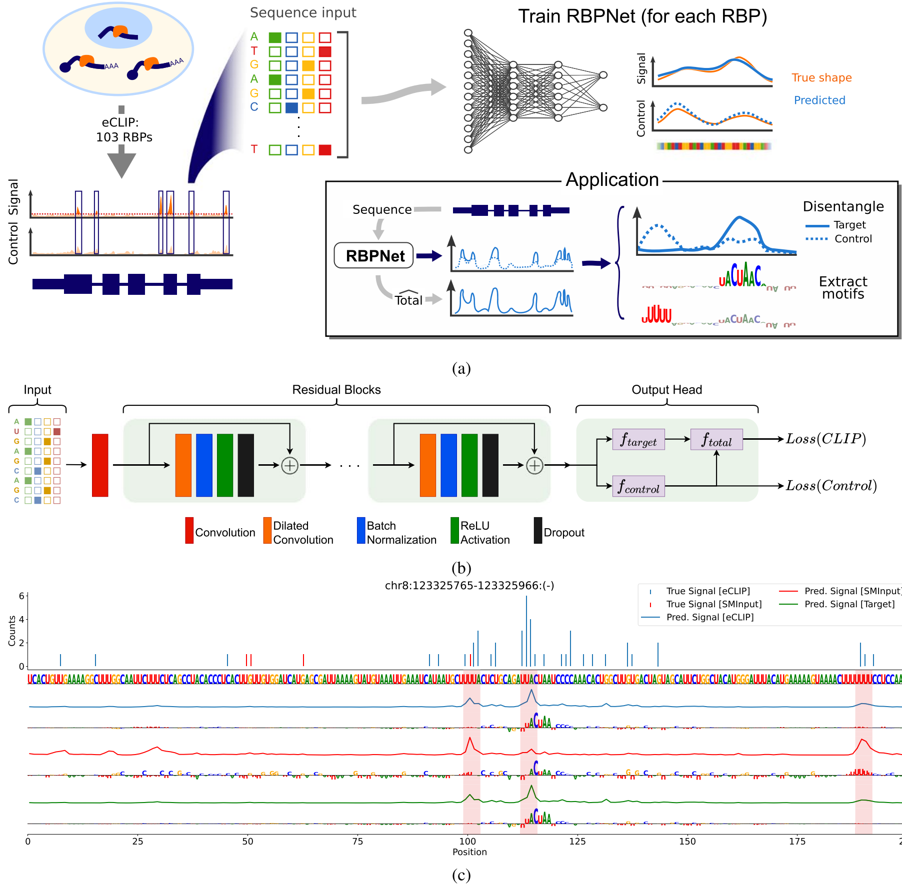


Fig. 1 RBPNet overview. **A** Schematic outline of data preparation, RBPNet training, and downstream applications. **B** RBPNet model architecture. The one-hot encoded RNA input sequence is first passed through a 1D convolutional layer, followed by several residual blocks, each consisting of a dilated convolution, batch normalization, ReLU, and dropout, respectively. Probability vectors of the *target* and *control* tracks are predicted from the output of the last residual block via a transposed convolutional layer while the *total* track is given by an additive mixture of *target* and *control* tracks. Given the predictions, a loss is computed by taking the sum of the negative log-likelihoods of the observed total and control counts. **C** Example prediction of an RBPNet model trained on eCLIP data of QKI showing observed counts (top) and predicted count distributions for the total (blue), control (red) and target (green) tracks. Integrated gradients feature attribution maps with respect to each predict track are shown below

interval, a negative log-likelihood loss is computed. In other words, the model is penalized in cases where it is unlikely that the observed crosslink counts were drawn from the distribution predicted by the model. RBPNet thus learns the shape of the crosslink count signal, which is subject to the RNA sequence under the assumption that RNA sequence composition is a driver of recognition (and subsequent binding) by RBPs. Similar to other CLIP-based protocols, eCLIP is known to be subject to experimental biases, for instance as a result of enhanced photoreactivity of single-stranded uridine (U) nucleotides during UV-radiation or contamination of eCLIP libraries with other RBPs [20]. Importantly, these biases are sequence-dependent and directly affect the distribution of cDNA truncation counts, hindering the identification of genuine sequence determinants of RBP binding. For that reason, eCLIP experiments are paired with a size-match input (SMInput) control experiment which omits the protein-specific immunoprecipitation

(IP) step, therefore capturing background crosslinking signal from other RBPs or technical biases. To prevent pattern learning of unspecific background and bias signal, RBPNet models the crosslink signal of the control experiment alongside the eCLIP signal. Specifically, RBPNet attempts to explain the observed eCLIP signal as a mixture of two signal components the *control* component, which is explicitly learned from the control experiment, and an unobserved *target* component, which represents the protein-specific signal (the "RBPNet bias correction" section). The *total* signal, that is, the count distribution of the eCLIP experiment, is then given by a weighted sum of the two components. This is illustrated schematically in the RBPNet output head in Fig. 1b as well as in Additional file 1: Fig. 1a, which outlines the network forward pass in the output head. The mixture of the two signals (target and control) is parameterized by a coefficient π, which is predicted from the RNA sequence and ranges between 0 and 1. Importantly, we control for technical bias in CLIP assays which we modeled with an additive mixture. This is in contrast to BPNet, a model for chromatin-immunoprecipitation assays which is dominated by DNA accessibility and sequence preference biases that were modeled as multiplicative noise [2].

RBPNet disentangles bias and protein-specific signal

The formulation of the *total* eCLIP signal as an additive mixture allows for disentanglement of the *target* from the *control* signal, where the predicted *target* signal represents in theory the bias-free, protein-specific crosslinking signal. This is exemplified in Fig. 1c, which shows the observed eCLIP and SMInput read start counts (top), as well as the *total*, *target*, and *control* signal predictions (bottom) using an RBPNet model trained on eCLIP data from the QKI RBP. The RBPNet *total track* (blue) captures well the experimental eCLIP read-start count profile, where the highest enrichment of eCLIP counts can be observed at position 113 of the sequence, immediately upstream of the known QKI binding motif (U)ACUUA [7]. Disentangling the predicted signal with RBPNet shows that the enrichment at this position is mostly attributed to the target, i.e., the true QKI-specific binding signal (green). On the other hand, the experimental eCLIP profile harbors two regions with lower enrichment of read start count around relative positions 102 and 189. Disentangling of the RBPNet total signal reveals that the count enrichment in these regions likely originated from experimental bias, as these regions coincide with elevated signal predictions of the control track (red). Further investigation of RBPNet predictions via Integrated Gradients (IG) [58] feature importance scores with respect to each signal track revealed that the known QKI binding motif (U)ACUUA [7] is correctly recovered in the IG map of the target track, corroborating the evidence that the predicted target signal shape corresponds to the bona fide QKI binding signal. In contrast, a degenerate QKI motif is observed in the IG maps of the control track, while the recovery of U-rich sequence motifs at the modes of the predicted control track distribution further strengthen the observation that those regions correspond to experimental bias. Note that while the SMInput experiment omits the immunoprecipitation, subsequent size-selection for the target protein still leads to a modest enrichment of protein-specific crosslink signal [65]. Therefore, the QKI-specific signal is partially captured in the bias track. While the predicted *total* track is a weighted average of *target* and *control* signal (with a mixing coefficient of 0.92, such that the target is dominating over the control track), genuine QKI binding signal is correctly recovered by the predicted *target* track. Likewise, signal enrichment mainly representing experimental bias are recovered by the *control* track, while being present in very low proportions in the *target* signal.

RBPNet predicts eCLIP signal shape at replicate-level accuracy

We next performed evaluation on 103 RBPNet models trained on data from ENCODE [62] eCLIP experiments. To this end, for each eCLIP experiment, candidate training pairs of 300 nt long sequences and crosslink count footprints were first generated without the use of a peak caller via lenient signal-thresholding (the "Methods" section) and subsequently split chromosome-wise into train, validation and test sets (the "Methods" section). Overall, we obtained an average of 302,752 candidate sites across eCLIP datasets, with a minimum of 7937 sites for LARP7 and a maximum number of 1, 105, 807 sites for HNRNPC. Subsequently, RBPNet models were trained separately for each RBP for at most 50 epochs, while the validation loss was observed for the purpose of early-stopping (the "Methods" section). Example predictions on hold-out samples with highest total observed counts for TIA1, QKI, and U2AF2 are depicted in Fig. 2a. Predictions show a high Pearson correlation coefficient (PCC) with the observed eCLIP counts (0.763, 0.830, and 0.872 for TIA1, QKI, and U2AF2, respectively), demonstrating that RBPNet can recapitulate the eCLIP signal shape at high accuracy. Next, we quantitatively assessed correlation of predicted and observed signal distribution. Figure 2b shows the average PCC of RBPNet (total track) predictions with observed eCLIP counts on hold-out samples versus the average PCC of counts between the two eCLIP replicates on the same sequence intervals. RBPNet achieves an average PCC between 0.200 (SSB) and 0.587 (HNRNPC), with an average PCC of 0.328. Strikingly, RBPNet prediction appear to outperform replicates, which have an average PCC of 0.149 across all RBPs. The reason for this are twofold. First, correlation between RBPNet predictions and observed counts are computed with respect to the merged count of both replicates, which reduces sampling effects and thus may increase PCC. Second, RBPNet predicts the count-generating distribution in the given interval, conditioned on the RNA sequence. In contrast, the observations in each replicate represent samples from the true (but unknown) count-generating distribution. As the estimated signal distribution by RBPNet approaches true distribution, the expected PCC between RBPNet predictions and a sample exceeds the expected PCC between the two samples (Additional file 1: Supplementary Text).

RBPNet enables whole-transcript inference and recovers single-nucleotide resolution binding sites

We next leveraged RBPNet's ability of performing prediction of RNA sequences of arbitrary length, despite being trained on fixed-length inputs. We explored first whether RBPNet can infer signal on entire transcripts by first selecting genes from GENECODE (Release 40) [14] from hold-out chromosomes and subsequently performing RBPNet predictions using models for all ENCODE RBPs. Figure 2f and g show RBPNet predictions (total track) on ENSG00000173207.12 and ENSG00000137955.15 for QKI and HNRNPC, respectively. Indeed, RBPNet predictions show a high correlation with the observed eCLIP counts (0.645 and 0.816, respectively), demonstrating that RBPNet

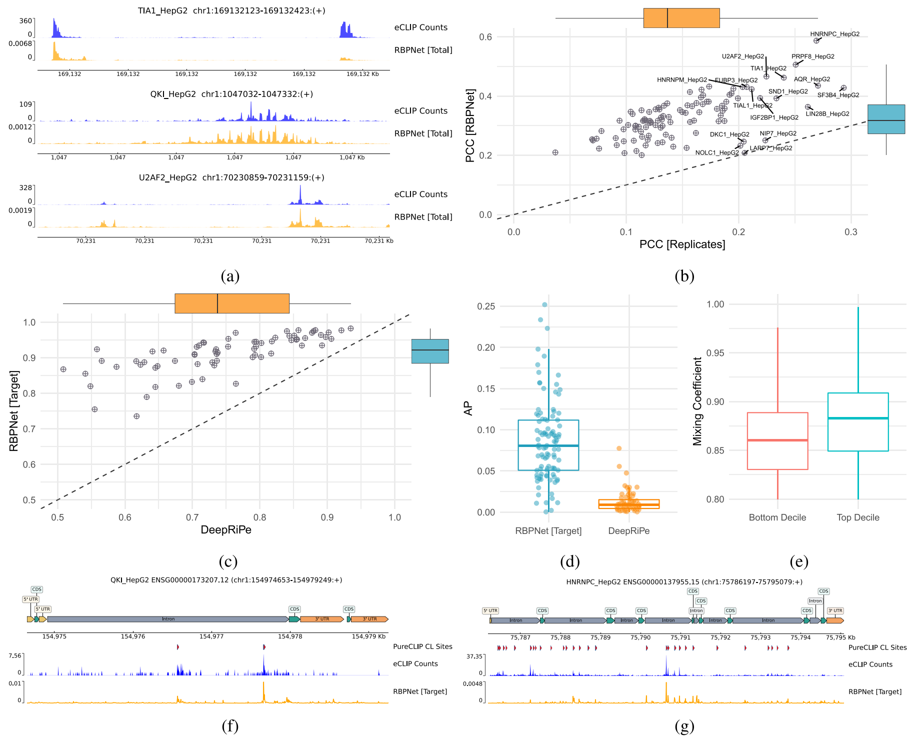


Fig. 2 RBPNet prediction performance on ENCODE eCLIP datasets. **A** RBPNet predictions (total track) on the highest-count hold-out samples for TIA1, QKI, and U2AF2. **B** Pearson correlation coefficient (PCC) of RBPNet predictions (total track) with observed eCLIP crosslink counts on hold-out samples vs. PCC of observed counts between the two eCLIP replicates. **C**, **D** Mean auROC and AP of RBPNet (target track) and DeepRiPe predictions with respect to crosslink and non-crosslink positions called by PureCLIP across transcripts from hold-out chromosomes, respectively. As pre-trained DeepRiPe models are available only for 70 (out of 103) ENCODE HepG2 RBPs, performance comparison is shown only for those RBPs. **E** Distribution of RBPNet mixing coefficients of the top and bottom decile ENCODE narrow peaks, sorted by eCLIP signal fold-change over the SMInput. High-affinity ENCODE narrow peaks show on average higher mixing coefficients compared to low-affinity peaks. **F**, **G** Example RBPNet (target track) whole-transcript predictions on ENSG00000173207.12 and ENSG00000137955.15, together with observed eCLIP counts and called PureCLIP peaks, for QKI and HNRNPC, respectively

models trained on rather short, fixed-size inputs generalize well to the task of whole-transcript prediction.

Given that RBPNet predictions show high correlation with observed eCLIP counts, we next assessed whether high-scoring RBPNet predictions coincide with peaks called by PureCLIP [38], a single-nucleotide peak caller that identifies significant crosslink sites from eCLIP and SMInput cDNA truncation counts using a Hidden Markov Model. To this end, we performed whole-genome peak calling with PureCLIP on ENCODE eCLIP datasets (the "Comparison with PureCLIP crosslink sites" section), identifying on average 46,459 crosslink (CL) sites per RBP, with a minimum and maximum number of CL sites of 1083 and 585,772 for NIP7 and HNRNPC, respectively. For each RBP, hold-out chromosome genes were intersected with PureCLIP CL sites and transcripts harboring at least 10 CL sites were selected for downstream evaluation. Subsequently, whole-transcript signal shape prediction was performed via RBPNet on selected transcripts. As PureCLIP peaks were called using both eCLIP and SMInput background signal information (the "Methods" section), predictions of the *target* track were used for RBPNet evaluation. For each transcript, auROC and average precision (AP) performance metrics were computed by treating positions at PureCLIP CL sites as positives and all other positions as negatives (Additional file 1: Table S1). Note that for this evaluation, PureCLIP peaks are considered to be the ground truth. Since RBPNet predicts the distribution of crosslink sites along genes, the probability at each position correlates with the genes length as well as the abundance of RBP binding on the transcript, rendering position-wise RBPNet predictions uncomparable across transcripts. Therefore, auROC and AP metrics are computed within transcripts and later averaged across transcripts in order to report the final, RBP-specific auROC and AP scores. In order to assess how the RBPNet behaves with respect to classification-based models, we next compared RBPNet predictions to DeepRiPe, a state-of-the-art classifier for prediction of protein-RNA interaction, by using a sliding-window approach to obtain pseudo single-nucleotide resolution scores (the "Methods" section). Since pre-trained DeepRiPe models for HepG2 ENCODE datasets were obtained directly from Ghanbari et al. [18], sequences in our hold-out set may have been present during training of DeepRiPe.

Figure 2c and d show the average auROC and AP scores, respectively. RBPNet outperforms DeepRiPe on all RBPs in terms of auROC performance, with an average auROC of 0.89 and a minimum and maximum auROC of 0.58 and 0.98 for SSB and HNRNPK, respectively. In contrast, DeepRiPe achieves a significantly lower average auROC of 0.74. Interestingly, RBP-wise auROC scores of RBPNet and DeepRiPe are strongly correlated ($PCC = 0.72$). This may suggest that some ENCODE eCLIP libraries are of lower quality or that RNA binding of some RBPs is more difficult to predict from sequence, possibly due to a lack of sequence binding motifs. Notably, RBPNet shows a lower variance of auROC performance across RBPs. Given that classification-based models such as DeepRiPe rely heavily on proper categorization of RNA sequences into binding and non-binding for training, this may indicate a higher robustness of RBPNet due to its training approach, which does not rely on upstream peak calling or labeling of RNA sequences. Both RBPNet and DeepRiPe AP values across RBPs range between 0.00032 and 0.2518 and .00038 and 0.0774, respectively, with RBPNet significantly outperforming DeepRiPe (average AP of 0.086 vs. 0.012, respectively). The generally low AP values of both methods is due to AP being sensitive to class imbalance, with the random AP baseline being equivalent to the fraction of positive samples in the dataset. Here, hold-out transcripts have an average PureCLIP CL site fraction of 0.0014 across RBPs, given that transcripts are expected to harbor orders of magnitude more non-CL than CL sites. Thus, while having low AP, RBPNet and DeepRiPe outperform the random baseline by a large margin. Overall, these results show that RBPNet is a powerful discriminator of PureCLIP CL and non-CL positions across the transcriptome, outperforming state-of-the-art classifiers.

RBPNet mixing coefficient captures relative eCLIP and SMInput signal abundance

RBPNet models the *total* eCLIP signal as a mixture of protein-specific *target* and *control* signal, weighted by a mixing coefficient which determines the fraction of *target* signal in the *total* signal (the "Methods" section). Additional file 1: Fig. 3a shows the distribution of average mixing coefficients on hold-out samples across eCLIP experiments, which range between 0.038 and 0.943 for proteins EXOSC5 and SFPQ, respectively, with a mean of means of 0.665. Furthermore, we observed that the distributions of mixing coefficients are generally uni-modal and narrow (Additional file 1: Fig. 4), suggesting that for the majority of experiments, the mixing coefficient is centered around a value that may be interpreted as an experiment signal-to-noise ratio. Indeed, the average mixing coefficient captures the similarity of eCLIP and SMInput experiment counts across hold-out samples ($PCC = -0.395$; Additional file 1: Fig. 3b), demonstrating that experiments with a lower predicted mixing coefficient (and thus a higher weight on the control track) show a higher similarity between the eCLIP counts and counts in the paired control.

To further evaluate whether the mixing coefficient captures the fraction of target and control signals, we inspected mixing coefficients of high-and low-affinity ENCODE narrow peaks [61]. To this end, for each RBP, we obtained ENCODE narrow peaks and ordered them decreasingly with respect to their log2 fold-change (logFC) of eCLIP signal over the SMInput. Peaks in the top and bottom deciles were then selected and extended up-and down-stream from their $5'$ end, which generally corresponds to the crosslink site, to yield 300 nt windows. Subsequently, RBPNet predictions were performed for each window and mixing coefficients were obtained. Figure 2e shows the distribution of mixing coefficients on top and bottom decile ENCODE narrow peaks. Indeed, top decile peaks receive on average significantly higher mixing coefficients compared to bottom decile peaks ($p < 2.2 \times 10^{-16}$), suggesting that the RBPNet mixing coefficient can separate high-affinity from low-affinity sites, where the latter contains a higher proportion of background signal.

RBPNet generalizes to iCLIP and miCLIP experiments

RBPNet may be trained on any genomic sequence with a corresponding nucleotide-wise count signal. To demonstrate that RBPNet generalizes to other CLIP-based protocols, we trained RBPNet on data derived from miCLIP and iCLIP experiments.

miCLIP enables in vivo identification of m6a RNA modifications at single-nucleotide resolution by incubating and subsequently crosslinking extracted RNA with a m6a-specific antibody [43]. After digestion of the covalently bound antibody, reverse transcription often truncates at a remaining polypeptide at the crosslink site, with pileups of truncation events yielding an m6a count signal across the transcriptome. miCLIP data from HEK293 and mESC cells was gathered from Kortel et al. [37] and processed similar to eCLIP data (the "Methods" section), before selecting candidate sites for training and evaluation (the "Methods" section). Subsequently, RBPNet models were trained on both cell lines using a similar architecture and hyperparameters as in eCLIP RBPNet models (the "Methods" section). While the mESC miCLIP experiment was paired with a knockout (KO) control experiment, no control experiment was available for HEK293. We therefore trained RBPNet on HEK293 miCLIP data without *target* and *control* modeling, that is, RBPNet was tasked to predict a single track describing the distribution of *total* miCLIP counts in the HEK293 cell line. RBPNet (total track) showed high signal correlation on miCLIP counts in both cell lines (Fig. 3a), reaching PCCs of 0.51 and 0.48 for HEK293 and mESC, respectively. To evaluate the ability of RBPNet to recover miCLIP single-nucleotide peaks, we again performed peak calling with PureCLIP (the "Methods" section), yielding a total of 2,011,704, and 278,311 for HEK293 and mESC miCLIP, respectively. We hypothesize

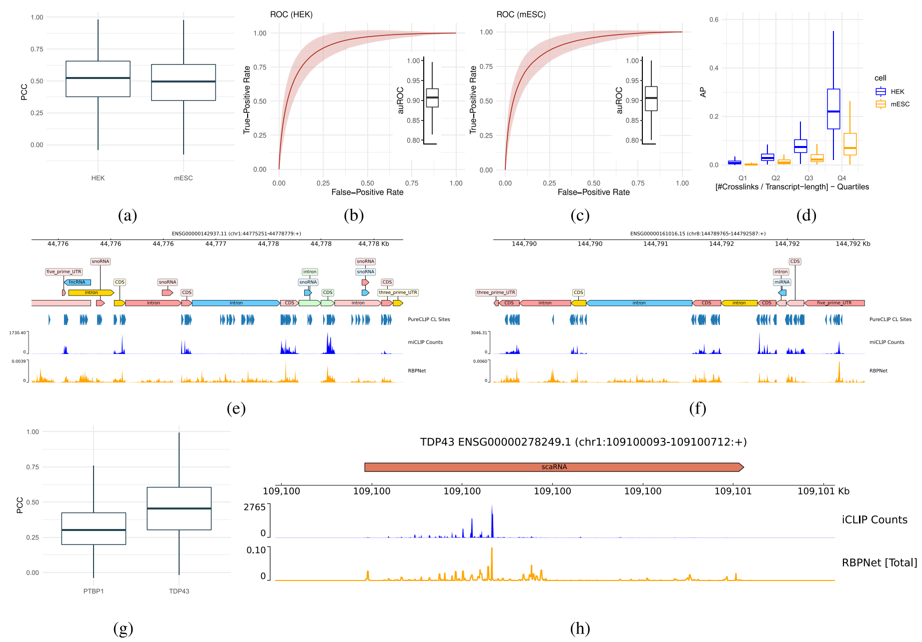


Fig. 3 Evaluation of RBPNet on iCLIP and miCLIP data. As in contrast to mESC, no control signal is used for m6a peak calling on HEK data, we consequently used RBPNet target and total track predictions for auROC and AP computation for mESC and HEK, respectively. **A** RBPNet PCC performance on hold-out samples of miCLIP experiments in HEK293 and mESC cell lines. **B**, **C** ROC performance of RBPNet whole-transcript predictions with respect to crosslink and non-crosslink positions called by PureCLIP for HEK293 and mESC cell lines, respectively. **D** AP performance on HEK293 and mESC in across transcripts for different PureCLIP crosslink site frequency quartiles. **E**, **F** Example RBPNet-HEK293 miCLIP predictions on ENSG00000142937.11 and ENSG00000161016.15. Notably, both predicted signal shape and observed miCLIP signal occurs predominantly in CDS and UTR regions. **G** Test set PCC performance on iCLIP experiments for PTBP1 and TDP43. **H** Predicted RBPNet signal shape for SCARNA2 (ENSG00000278249.1)

that the significantly higher number of PureCLIP CL sites in HEK293 may be due to a lack of control signal, which in context of mESC may lead to a large number of candidate CL sites being discarded. While the KO control is used for PureCLIP background normalization for the mESC cell line, no control is used for peak calling on the HEK293 cell line, as this was not available. Due to the lack of controls, PureCLIP yields a significantly higher number of CL sites in HEK293 compared to mESC, where a high portion of them may correspond to false positives or noise. Similar to PureCLIP analysis in eCLIP, we then performed whole-transcript inference of miCLIP signal shape on genes of the hold-out set. Figure 3b and c show the ROC curves for HEK293 and mESC, respectively. RBPNet performs well in terms of auROC on both cell lines, with an average auROC of 0.89 for HEK293 and 0.88 for the mESC cell line. In contrast, RBPNet achieves an AP scores of 0.1 and 0.043 for HEK293 and mESC, respectively, with AP performance naturally increasing together with the fraction of positions harboring PureCLIP CL sites. This is illustrated in Fig. 3d, which shows the distribution of AP scores across transcripts grouped into quartiles based on their CL fraction. We consequently observe a lower AP score for mESC compared to the HEK293 cell line as a result of the lower number of PureCLIP CL sites in mESC. Figure 3e shows example RBPNet predictions (total track) on two human genes (ENSG00000142937.11 and ENSG00000161016.15, respectively) from the hold-out chromosomes using the HEK293 RBPNet model. Interestingly, RBPNet predictions (as well as observed signal) predominantly occur in coding regions (CDS) and 5'/3' untranslated regions (UTRs), while being only slightly present in introns, which is in line with previous study reporting that m^6A methylation mainly those genomic regions [33, 45].

We next evaluated the performance of RBPNet on individual-nucleotide CLIP (iCLIP) data [36]. Compared to eCLIP, it makes use of extra circularization and linearization steps, which allow all cross-linked cDNA fragments to be amplified and sequenced, and a quality control step to assess specificity of pulled-down protein-RNA complexes. To this end, we gathered iCLIP data from Hallegger et al. [23] and Haberman et al. [20] for TDP43 and PTBP1 proteins, respectively, which was processed as described in the "Data and preprocessing" section. As no paired control experiment was available for either RBP, we again trained RBPNet by omitting modeling of *target* and *control* signal and instead tasked RBPNet with predicting the total iCLIP count distribution directly. Figure 3g shows the distribution of PCCs on test set samples for the PTBP1 and TDP43 models. RBPNet reaches an average PCC of 0.46 for TDP43 and 0.320 for PTBP1. Notably, RBPNet reaches a comparable PCC of 0.366 on the PTBP1 eCLIP dataset, which may suggest that the fraction of PTBP1 signal (and thus RNA-binding) explained by RNA sequence is comparable in the eCLIP and iCLIP datasets. This demonstrates the capability of RBPNet to achieve high predictive performance of the RBP-RNA interaction signal shape, independently of the protocol used to generate the data the model is trained on. In order to qualitatively assess the ability of the RBPNet-TDP-43 model to predict signal shapes on full length transcript when trained on iCLIP data, we manually investigated the prediction profile on the hold-out transcript with the highest absolute counts. Figure 3h shows TDP-43 RBPNet predictions and observed iCLIP counts for ENSG00000278249.1 (SCARNA2), a scaRNA associated with DNA repair pathway regulation that has been previously described to be interacting with TDP43 [5, 30]. Indeed, the profile predicted by RBPNet strikingly reflects the observed signal.

Sequence attribution maps capture RBP-binding motifs

Recognition of target RNAs by RBPs is in part driven by local sequence features, also known as binding motifs. The identification of RBP-binding motifs is crucial for understanding RBP target recognition and the regulatory grammar present in RNA sequences. While deep learning models were long regarded as black boxes, recent feature attribution methods, such as Integrated Gradients (IG) [58], allow for the identification of input features that contributed significantly to the observed model prediction. In the context of RBPNet, these methods "attribute" a given prediction to nucleotides in the input RNA sequence by assigning a score to each position. Nucleotides that were primarily responsible for the observed crosslink count distribution, such as those residing in binding motifs, receive a higher score compared to nucleotides that did not contribute towards protein-RNA crosslinking. IG attribution maps may be computed with respect to any of the three output track, i.e., control, target and total. Attributions of the target track are

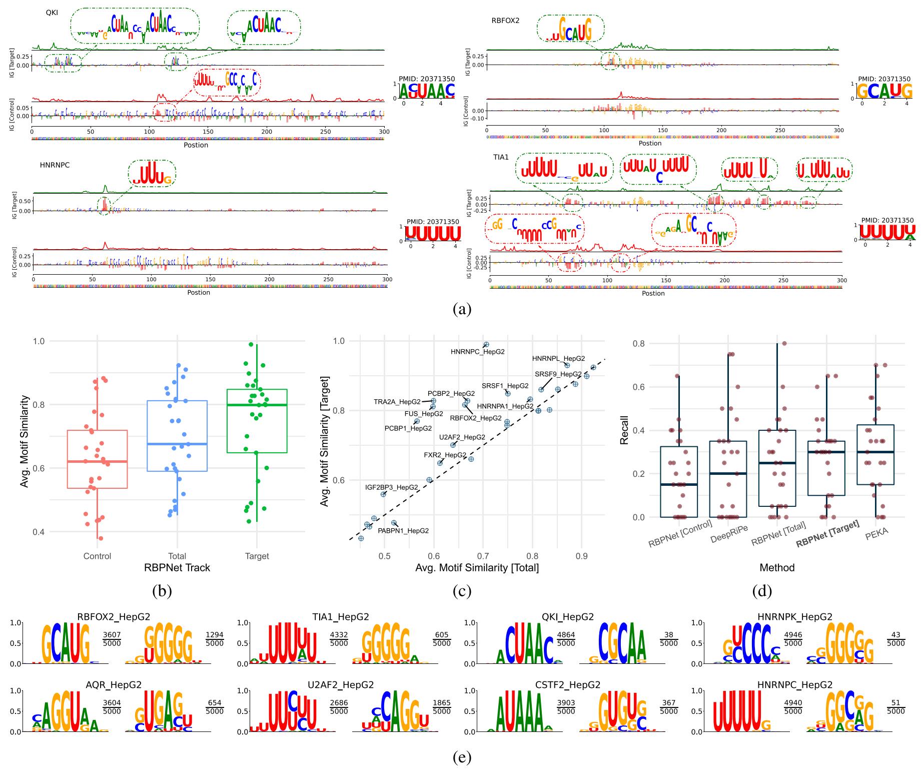


Fig. 4 RBPNet feature attribution maps and binding motif discovery. **A** Example integrated gradient attribution maps with respect to the *target* track for RBFOX2, HNRNPK, TIA1, and QKI with corresponding motifs taken from the RBPmap database. **B** Distributions of similarity scores between 5-mers extracted from RBPNet attribution maps and PWMs of motifs reported in RBPmap for *control*, *target*, and *total*. **C** Average RBPmap PWM similarity across RBPs for 5-mers extracted from *target* and *total* track attributions. While similarities of the *target* track on par or higher compared to the *total* track for the vast majority of RBPs, the improvement is more pronounced for some RBPs. **D** Recall of the top 20 in vitro 5-mers recovered by the top 20 5-mers extracted from attribution maps of RBPNet *control*, *target*, and *total* tracks as well as DeepRiPe and the PEKA motif finder. RBPNet *target* track and PEKA show comparable performance, outperforming DeepRiPe and the RBPNet *total* track. **E** Consensus motifs constructed from extracted 5-mers of the RBPNet *target* track. Consensus motifs with highest (primary) and second highest (secondary) k-mer support are shown. The corresponding k-mer support is shown as a fraction of the total number of extracted 5-mers next to the consensus motif logos

expected to highlight nucleotides that contributed significantly towards protein-specific crosslinking, while attribution of the control track are expected to explain the unspecific background and bias signals[1].

Figure 4a shows examples of IG attribution maps, computed with respect to *control* and *target* tracks for test-set samples using eCLIP RBPNet models for RBFOX2, HNRNPK, TIA1, and QKI, alongside PWMs of consensus motifs reported in literature, obtained from the RBPmap database [50]. The corresponding predicted *control* and *target* signals are shown above the attribution maps (in red and green, respectively). Target track IG attribution maps show the presence of highly predictive sub-sequences that correspond to known binding motifs for each of the shown RBPs. For instance, IGs of

[1] Note that not all technical CLIP bias is manifested in the RNA sequence and thus learnable by the RBPNet model. Therefore, only the sequence-component of the bias will be captured in control track IG maps.

HNRNPC show the characteristic U-motif, while three distinct canonical ACUAAC motifs can be seen in the IGs of QKI. In contrast, control track attribution maps do not show the presence of clear binding motifs, with a general attribution score enrichment at G and C nucleotides. We next performed a global quantitative assessment of how well RBPNet attribution maps can recover known binding motifs across RBPs in the ENCODE eCLIP database. To this end, we selected the top 5000 ENCODE narrow peaks for all ENCODE eCLIP experiments with trained RBPNet models and computed IG attribution maps with respect to *target*, *control*, and *total* tracks for each RBP. For each attribution map, we extracted 5-mers with highest sum IG scores (the "Methods" section). Extracted 5mers were then compared with position-weight matrices (PWMs) of literature motifs obtained from the RBPmap [50] database by computing the similarity between each 5mer and its corresponding RBPmap motif (the "Methods" section). In total, 29 out of the 103 ENCODE RBPs with RBPNet models were represented with at least one PWM in RBPmap, which were then selected for downstream analysis. Figure 4b shows the distribution of average similarity scores of extracted 5mers to RBPmap PWMs across RBPs for each of the three RBPNet prediction tracks. 5-mers extracted from attribution maps of the *target* track show the highest RBPmap-similarity, consistent with the fact that this track represents the de-biased, protein-specific signal predictions, followed by the *total* and *control* tracks, respectively. Notably, on average 5-mers extracted with respect to the control track have the lowest similarity to known RBP PWMs, highlighting the ability of RBPNet to extract different sequence representation for true signal and bias, respectively. To investigate the degree of improvement that signal de-biasing in the *target* track offers over the *total* track for individual RBPs, Fig. 4c shows the average *target* and *total* track RBPmap-similarity for each RBP. Interestingly, we find that while for some RBPs, the *target* track leads to no or only modest improvements of RBPmap-similarity compared to the *total* track, other RBPs, such as HNRNPC, PCBP2, RBFOX2, and TRA2A, appear to benefit strongly from the signal de-biasing of the *target* track. This may reflect variable levels of experimental bias across eCLIP datasets.

RBPNet IG attribution maps recover in vitro binding motifs

In vitro experiments on protein-RNA interactions, such as RNA-Bind-N-Seq and RNA-compete, offer an orthogonal view to validate motifs identified from in vivo data, as they do not harbor crosslinking specific biases or contamination of experiments with other RBPs and therefore measure intrinsic affinity of RBPs to RNA in an unperturbed environment [41, 52]. To examine whether modeling of the control signal as an auxiliary task can increase the specificity of predicted CLIP signal, we cross compared 5-mers previously obtained from RBPNet IG attribution maps of *target*, *control*, and *total* tracks with 5-mers enriched in corresponding RNA-Bind-N-Seq (RBNS) or RNAcompete in vitro datasets. In total, we evaluated 27 eCLIP datasets in the HepG2 cell line, for which either RBNS (16 RBPs) or RNAcompete (11 RBPs) data was also available.

To measure the agreement between in vitro 5-mers and RBPNet 5-mers obtained in the previous section, we first computed the sum of IG_{sum} scores for each unique 5-mer as a measure of importance with respect to CLIP signal shape prediction. We then calculated the RBPNet recall for each track by taking the fraction of the top 20 in vitro 5-mers that were recovered in the top 20 5-mers by RBPNet on eCLIP datasets (the "Methods" section). We found that across evaluated RBPs, the RBPNet *target* track recovered a significantly higher proportion of relevant in vitro 5-mers from eCLIP than the *total* track, suggesting that RBPNet can successfully increase the specificity of eCLIP signal (Fig. 4d). As expected, the *control* track recovered the least in vitro k-mers; however, for some RBPs, even the *control* track alone could retrieve high ranking in vitro motifs. This effect could be explained by a partial enrichment of RBP-specific signal in the control experiment, as suggested by a previous study, which evaluated the effect of using eCLIP narrow peaks in contrast to SMInput-agnostic peak-calling on discovery of relevant binding motifs from eCLIP data [39]. Next, we set out to assess whether the governing sequence features learned by RBPNet could be reliably used for motif discovery. To this end, we compared the RBPNet recall to positionally-enriched k-mer analysis (PEKA), a state-of-the art tool for discovery of enriched k-mers from individual CLIP datasets [39] and DeepRiPe. In contrast to RBPNet, PEKA models background from intrinsic crosslinking signal and does not use SMInput controls; therefore, it provides valuable orthogonal view in how specificity of the motifs can be impacted by different background modeling approaches. Surprisingly, we found that the RBPNet *target* track recovered a similar proportion of relevant in vitro k-mers compared to PEKA, despite the fact that RBPNet was not originally designed for objective of motif discovery (Fig. 4d). Lastly, we investigated whether a direct modeling of eCLIP signal, compared to binary labels (bound/unbound) assigned to entire RNA sequences, offers an advantage in the context of motif discovery. To this end, we compared RBPNet recall performance to DeepRiPe by extracting 5-mers from DeepRiPe attribution maps (the "Methods" section). Indeed, both RBPNet *target* and *total* track outperform DeepRiPe in terms of in vitro k-mer recovery (mean recall of 0.294, 0.254, and 0.222, respectively), suggesting that direct modeling of raw eCLIP signal improves binding motif discovery irrespectively of bias modeling. Furthermore, these results indicate that enriched in vitro k-mers are strong predictors of true CLIP signal and that RBPNet learns to associate the presence of these k-mers with high count signal. Lower agreement of the RBPNet *total* track (compared to *target* track) with in vitro k-mers suggests that a fraction of the *total* track signal is explained by sequence features not present in in vitro k-mers, which likely correspond various possible technical sources of sequence biases or contaminant signals in CLIP [22, 39].

Consensus motifs reveal primary and secondary motifs

Having demonstrated that RBPNet successfully identifies predictive k-mers that coincide with in vitro datasets, we next constructed consensus binding motifs to concisely represent the sequence binding preference of each eCLIP RBP. To this end, we select 5-mers previously derived from RBPNet *target* track attribution maps and build consensus motifs by successively aligning 5-mers in the order of highest to lowest IG importance (the "Methods" section). Figure 4e shows consensus motifs together with the fraction of supporting 5mers for selected RBPs. A full list of consensus motifs extracted with RBPNet is displayed in Additional file 1: Fig. 2 and Additional file 2: Table S2. To investigate which motifs derived from RBPNet are known and which ones are novel, we compared RBPNet motifs with motifs reported across existing databases, including ATtRACT [19], RBPDB [8], mCross [11], oRNAment [4], and RBPmap [50]. Additional file 2: Table S2 depicts RBPNet consensus motifs together with their similarity to motifs reported for the same protein in each of the five evaluate databases (the "Methods" section). Primary binding motifs identified with RBPNet agree with previously identified motifs, including RBFOX2, QKI, TIA1, U2AF2, and HNRNPC. We further identify secondary consensus motifs, which may represent alternative binding preferences or co-factor binding (the "Methods" section). In addition, RBPNet suggests novel candidate motifs for several RBPs, including AQR, SAFB, WDR43, and NIP7, for which no motifs exist in any of the evaluated databases. Several RBPs, including RBFOX2, TIA1, and (to a lesser extend) HNRNPK, show a G-rich secondary motif (Fig. 4e), with some showing G-rich primary motifs (BCCIP and CSTF2T; Additional file 2: Table S2). We hypothesize that this may be due to co-factor binding of an RBP with G-motif preference or, as suggested by previous studies, may indicate contamination of eCLIP libraries with another RBP [39, 62]. For this reason, G-rich motifs are to be treated with care, as they may (in general) not represent genuine binding preference of the target RBP. A complete list of consensus motifs derived from RBPNet are shown in Additional file 1: Fig. 2.

Studying the impact of sequence variants on protein-RNA interaction with RBPNet

RBPs are highly evolutionary conserved and have been associated with an abundance of human diseases, particularly in the context of degenerative disorders [9, 49]. Recently, Gebauer et al. [17] found that over 1000 RBPs are mutated in context of disease, which amounts to > 20% of proteins annotated with disease-associated mutations. Besides altered coding sequences of RBPs, nucleotide polymorphisms in their RNA targets may impact transcript regulation via loss of binding sites. Indeed, Park et al. [49] showed that dysregulation of RNA target sites from RBPs with a diverse set of functions represents is a key driver of psychiatric disorder risk. Therefore, computationally quantifying the impact of variants with respect to protein-RNA interaction at large scale is important for the prioritization of causal variants in context of disease. Classification-based methods have previously been utilized for the scoring of variants by taking the difference in predicted binding probability between the reference and alternative alleles. In contrast, RBPNet predictions are vector-valued, which complicates a direct comparison of both alleles. Here, we investigated the ability of RBPNet to score the effects of sequence variation on protein-RNA interaction by quantifying variant impact as the divergence between the predicted CLIP-seq (target) signal distributions of the reference and alternative alleles. We demonstrate that resulting impact scores can prioritize variants that may be associated with the disruption of protein-RNA interaction.

Figure 5a exemplifies our variant scoring approach on *rs6981405*, an A-to-C transversion within the *DDHD2* gene that disrupts QKI binding and which has been associated with schizophrenic risk [49]. As shown, in silico mutagenesis leads to change in predicted RBPNet target signal around the variant, with a lower signal amplitude in the alternative (ALT) allele compared to the reference (REF), which we quantify as the Kullback-Leibler (KL) divergence between REF and ALT predictions (the "Methods" section), yielding a scalar impact score. Feature attribution maps (Fig. 5a, bottom) of the REF and ALT predictions reveal that *rs6981405* disrupts the well known QKI binding motif ACUAAC [21]. The strongly negative implications of the A-to-C transversion at the last position of the binding motif is correctly detected by RBPNet, as is evident

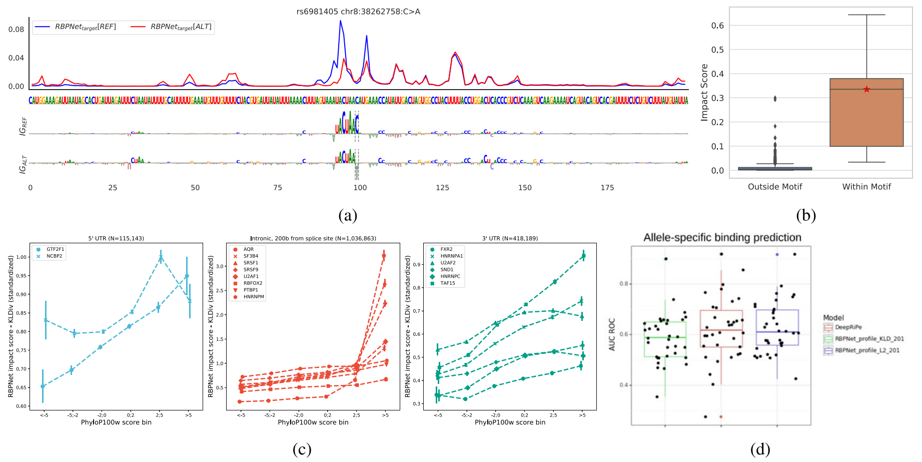


Fig. 5 RBPNet variant impact prediction. **A** Impact of the *rs6981405* variant on the predicted RBPNet *target* signal. Reference (blue) and alternative (red) signal is shown in a 200-nt window around the variant position. The corresponding feature importance maps for reference and alternative sequence are shown below. A C-to-A transversion at the 5′ end of the QKI binding motif leads to a drastic change of the predicted signal compared to the reference signal, which can be quantified as the KL divergence between the reference and alternative allele predictions. **B** Comparison of impact scores of a systematic in silico mutagenesis of each position towards each of the 3 alternative bases within in a 200-nt sequence window around *rs6981405*. **C** RBPNet impact scores (standardized) on gnomAD variants as a function of sequence conservation, measured via the PhyloP score. Mutations are separated per biotype and their impact is evaluated for RBPs known for biotype-specific activity. Impact scores are higher in regions that are under strong evolutionary constraints. **D** Classification performance of allele-specific binding variants by RBPNet, measured in KLD and L2 norm (the "Methods" section), and DeepRiPe

from the change of high-positive to high-negative attribution at the mutation site. To evaluate whether greater impact is assigned to SNPs within the QKI motif compared to SNPs outside the motif, we performed a systematic in silico perturbation analysis of each nucleotides within a 200-nt window around the *rs6981405* SNP. Figure 5b depicts the distribution of a total of 600 variant impact scores, grouped based on whether they reside within or outside the QKI binding motif. Indeed, the majority of non-motif perturbations lead to small changes in predicted signal profile and thus to small impact scores, while mutations falling within the QKI motifs lead to significantly larger impact scores. Further examples are given in Additional file 1: Fig. 7. This demonstrates that KL divergence is an appropriate metric for scoring the impact of sequence variants based on RBPNet predictions.

RBPNet variant impact scores are higher in evolutionary constrained regions

To assess whether RBPNet impact scores can identify the disruption of regulatory RNA elements, we investigated the elevation of impact scores in non-coding regions which are under negative selective pressure. To this end, we obtained a set of 1,570,195 SNV from gnomAD [31] which were scored with RBPNet models of a representative set of 15 RBPs, chosen across a broad range of post-transcriptional pathways (the "Methods" section), in order to keep the scoring of variants in a computationally feasibly range. Figure 5c depicts the average impact score (measured in terms of KL divergence) as a function of sequence conservation, measured by PhyloP score across 100 vertebrates, at the variant position. Indeed, impact scores are strongly correlated with sequence conservation across all investigated RBPs, suggesting that RBPNet identifies the disruption of conserved regulatory RNA elements that engage in protein-RNA interaction. In addition, we investigated the correlation of impact scores with variant allele frequencies (AF) (Additional file 1: Fig. 8). Surprisingly, no clear association between RBPNet KLD impact scores and variant AF could be identified. We speculate that fitness constraints on RNA regulatory motifs may not be prominent enough on population scale, which may render the comparison of impact scores with variant allele frequency inconclusive. In addition, constraints are likely biotype-specific, requiring more sophisticated stratification of non-coding variants which are under negative selection [12].

Investigating allele-specific binding (ASB) with RBPNet

We next investigated whether RBPNet impact scores can stratify variants that are associated with allele-specific CLIP-seq signal enrichment from a set of background variants. To this end, we obtained and processed variants associated with allele-specific binding (ASB) from Yang et al. [66] (the "Methods" section). Qualitative evaluation of ASB sequences with RBPNet revealed that variants in close proximity may result in false-positive ASB variants, as illustrated in Additional file 1: Fig. 6, which shows 4 ASB-associated SNVs for QKI within a 200-nt interval. RBPNet prediction of each variant allele revealed that only 2 (top left/top right) out of 4 SNVs are associated with a disruption of the QKI-binding motif, which induces substantial changes in the predicted RBPNet signal. The two remaining SNVs (bottom left/bottom right) fall outside a QKI binding motif and thus show only a negligible change of predicted RBPNet signal. KLD impact scores for SNVs impacting QKI motifs are considerably higher (0.28 and 0.12) compared to SNVs outside of motifs (0.05 and 0.02), which is in line with the strong dependence of allele specific effects of SNVs with respect to their distance to known motifs [66] and suggests that variant impact analysis via RBPNet may aid in prioritizing causal SNVs for allele-specific binding.

In order to deplete the set of ASB-associated SNVs for false-positives, SNVs in close proximity ($< 300nt$) were removed from the ASB set. Figure 5d shows the distribution of auROC scores for RBPNet and DeepRiPe [18] with respect to separating ASB and non-ASB SNVs via impact score. While RBPNet can separate ASB from non-ASB SNVs via KL divergence impact scoring ($auROC = 0.59$), DeepRiPe shows a higher performance ($auROC = 0.62$). Interestingly, further evaluations of different RBPNet impact scoring metrics revealed that comparison of reference and alternative allele prediction via the L2-norm of the element-wise difference between the two vectors, yields a performance on-par with DeepRiPe ($auROC = 0.62$). Furthermore, Additional file 1: Fig. 5 shows that performance of RBPNet and DeepRiPe is RBP-specific, suggesting that these methods may complement each other. As several metrics for comparing two vector-valued predictions exist, this suggests that the variant impact scoring of RBPNet may be further improved via systematic metric search and that this metric may be task-specific.

RBPNet discriminates between functional and non-functional mutations nearby splice junction

Splicing is a complex and tightly regulated post-transcriptional processes in higher eukaryotes that requires concerted binding of multiple RBPs via spliceosomal

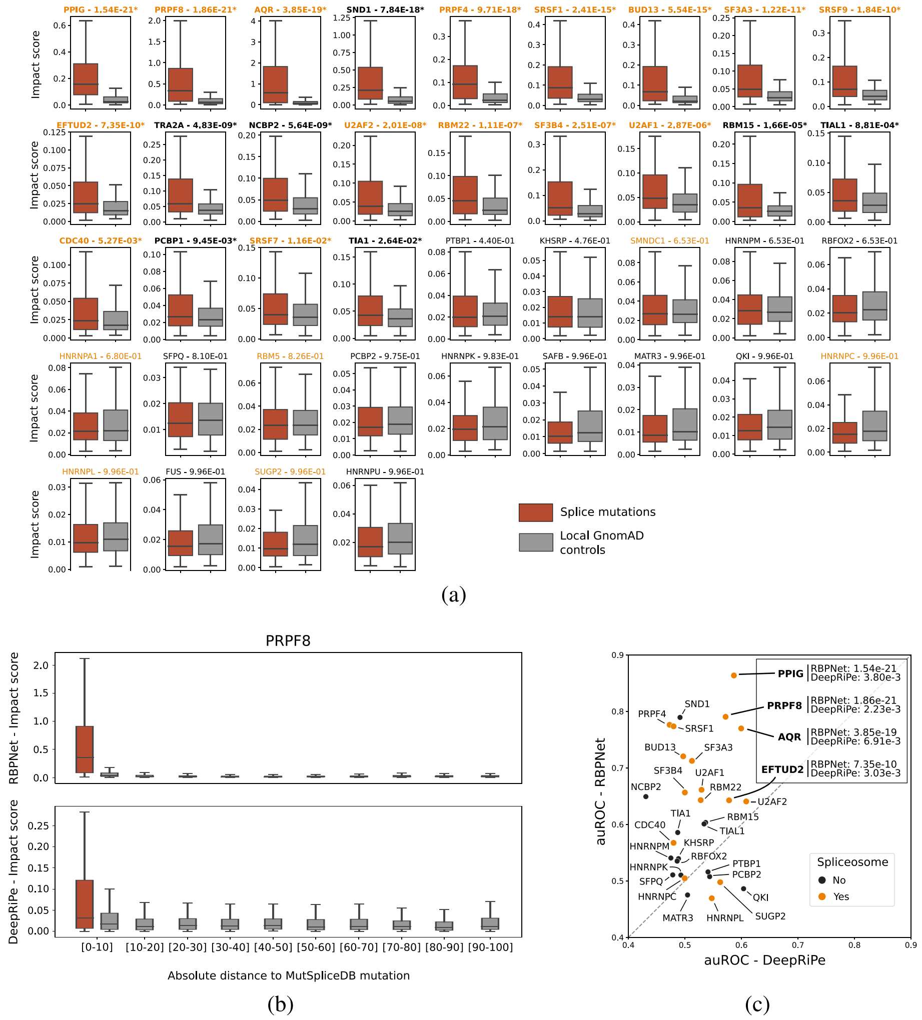


Fig. 6 Scoring of 232 splicing mutations from MutSpliceDB along with 6087 control mutations from gnomAD taken in their vicinity, using 40 splicing-related RBPNet models. **A** Boxplots of RBPNet impact scores from splicing mutations and local controls per RBP. The 40 RBPs are ordered by their adjusted (Benjamini/Hochberg) *P*-values from Wilcoxon signed rank tests. Title in bold: RBPs with significant *P*-values at $\alpha = 0.05$ (22/40); orange font color indicates the spliceosomal RBPs. **B** Impact score distribution for splice mutations (red boxplot) and gnomAD control mutations (gray boxplots) per absolute, relative-distance bin, with impact scores obtained from RBPNet and DeepRiPe models for the spliceosomal RBP PRPF8 (being the most significant model from DeepRiPe following the Wilcoxon signed rank tests). **C** Scatterplot of area under the receiver operating curve (auROC) calculated from RBPNet models (y) and DeepRiPe models (x). All 30 RBPs annotated as splicing-related in common are depicted, with spliceosomal RBPs in orange. Four RBPs are specifically highlighted for being the only models showing significant *P*-values from the Wilcoxon signed rank tests applied on DeepRiPe models, and their adjusted *P*-values are reported (along the adjusted *P*-values for RBPNet)

complexes, with sequence variants disrupting regulatory binding motifs being associated with severe deleterious effects [15]. To evaluate RBPNet's ability to score the impact of sequence variants on RNA-binding of splicing factors, we obtained a set of 232 splicing-associated mutations from MutSpliceDB [47]. Indeed, we observed greater impact scores for splicing mutations compared to their local controls for 22 out of 40 splicing-related RBPs (Fig. 6a, the "Methods" section). Notably, the set of significant RBPs showed an over-representation of spliceosomal RBPs (15 out of 22), concordant with

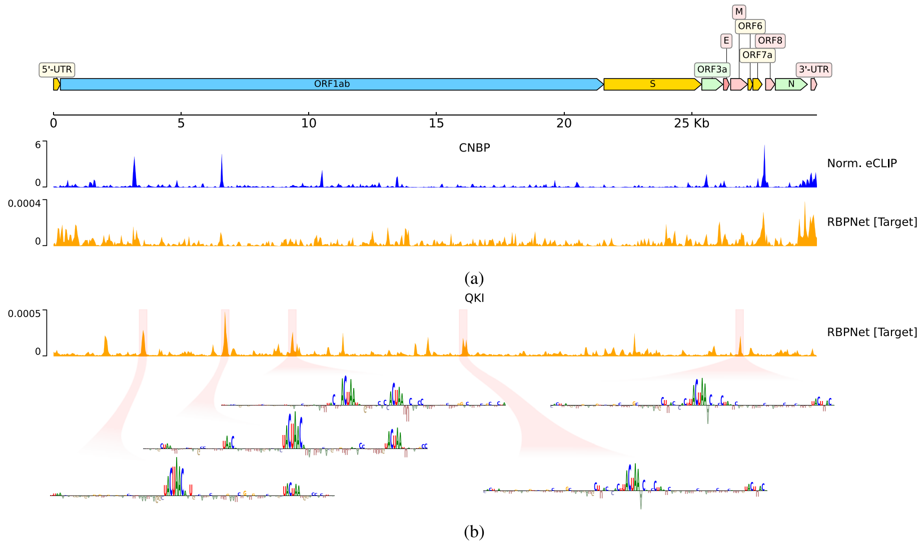


Fig. 7 Predicting SARS-CoV-2 binding with RBPNet. The top shows target track predictions using a model trained on CNBP eCLIP data, together with the normalized observed signal taken from Schmidt et al. [55]. Shown below are predictions from a RBPNet-QKI model trained on ENCODE eCLIP data, as well as IG attribution maps around the top-5 positions with highest predicted probability

their higher susceptibility to be directly impacted by splicing mutations at splice junctions. Applying the same procedure with DeepRiPe models, only 4 models (out of 30) showed a significant difference in impact scores allowing discrimination between splicing mutations and local negative controls. This is illustrated by the distribution of scores at various relative distance from splicing junctions, as seen for example with the pre-mRNA processing factor 8 (PRPF8, Fig. 6b), a core protein of the spliceosome. While the DeepRiPe model showed the most significant discrimination between functional and non-functional mutations (adjusted P-value $= 2.23 \times 10^{-3}$), we can see that the score distribution from non-functional mutations at very short distance (< 10 nt) is slightly more elevated than for mutations beyond. On the other hand, the RBPNet models shows a very clear discrimination even at such short distance. We confirmed the improvement of RBPNet over DeepRiPe for scoring variants by comparing the area under the Receiver Operating Curves from each pair of models (Fig. 6c), showing that RBPNet was a better classifier for 26 of the 30 models in common.

Leveraging RBPNet to infer human RBP binding on viral RNAs

We lastly evaluate RBPNet's ability to infer missing eCLIP signal shape on foreign RNAs. Viruses, including SARS-CoV-2, extensively interact with the host's RBPome [13, 16, 46]. As the large-scale experimental identification of human RBP binding on viral RNA is associated with significant monetary and labor costs, computational imputation of binding sites represents an attractive alternative to identify crucial host factors involved in the virus life cycle. Recently, Labeau et al. [40] experimentally identified binding of QKI to SARS-CoV-2 and reported that QKI-knockout cells are less permissive than wild-type cells.

Here, we utilize RBPNet models trained on eCLIP datasets to predicting binding profiles of human RBPs to SARS-CoV-2 at single nucleotide resolution. Figure 7b shows

RBPNet target track predictions for QKI, showing several high-scoring regions across the SARS-CoV-2 sequence. IG attribution maps at sequence windows around the top 5 highest prediction scores revealed the presence of the QKI-binding motif, suggesting that the RBPNet-QKI model predicts bona fide binding profiles. To further validate that RBPNet predictions on SARS-CoV-2, we obtained eCLIP data for CNBP from Schmidt et al. [55]. Indeed, RBPNet target track predictions correlate significantly with the control-normalized eCLIP signal (Fig. 7a), with an PCC of 0.210. Together, this indicates that RBPNet trained on human eCLIP experiments may be used to extrapolate eCLIP signal shapes to non-human RNA sequences.

RBPNet webserver

For ease of use, a *BioLib* webserver version of RBPNet is accessible at https://biolib.com/mhorlacher/RBPNet, enabling prediction of signal profiles and sequence attribution maps with pre-trained models used in this study on user-provided RNA sequences.

Discussion

While high-throughput single-nucleotide CLIP-seq methods offer unprecedented insights into the protein-RNA interaction landscape of RNA-binding proteins, they are limited to transcripts expressed in the experimental cell type at the time of the experiment. Therefore, researchers must rely on computational methods to impute missing binding information on unexpressed transcripts and or foreign RNAs from sequence. In recent years, an abundance of machine learning methods have been developed for the prediction of protein-RNA interaction from RNA sequence, with the most recent iteration of methods relying on deep neural networks to achieve high state-of-the-art performance. However, current classification-based methods assign predicted binding probabilities to the entire input sequence, typically in the order of hundreds of nucleotides, creating ambiguity with respect to the exact location of protein-RNA interaction. Furthermore, the binary labels used in context of classifiers may only contain a fraction of biological information generated by CLIP experiments, which may hinder the learning of complex associations between RNA sequence and RBP-binding. The implicit common goal of protein-RNA prediction methods is a near perfect recapitulation of binding information offered by their in vivo experimental counterparts, which generate nucleotide-wise counts signal across the transcriptome. In this study, we presented a significant milestone towards this goal in the form of a novel deep learning model, RBPNet, which predicts the CLIP-seq signal shape from RNA sequence at single-nucleotide resolution. By training and evaluating RBPNet on eCLIP, miCLIP and iCLIP datasets, we demonstrated that the model is able to predict the experimental signal shape at high accuracy, reaching replicate-level performance and is applicable to data from different CLIP-based protocols. We leveraged RBPNet's ability to processes sequences of arbitrary length at prediction time and impute signal shape on entire gene sequences, some of hundreds of kilo-bases in size, and showed that high-scoring positions predicted by RBPNet coincided with single-nucleotide peaks identified from experimental data via the PureCLIP peak caller. Due to their fundamentally different outputs, establishing unbiased comparisons between RBPNet and classification-based models is challenging, as the later assigns predictions to sequence windows rather than individual positions by design.

To enable comparison with DeepRiPe, a state-of-the-art CNN classifier, we generated pseudo position-wise scores via a sliding window approach and showed that RBPNet significantly outperforms DeepRiPe at recapitulating of PureCLIP crosslink sites. Note that comparison of classifiers (such as DeepRiPe) and methods that directly model the raw experimental signal at nucleotide resolution (such as RBPNet) should be interpreted with care. As RBPNet is the first method to model the nucleotide-wise distribution of CLIP-seq signal as a function of sequence, we compared RBPNet to a current state-of-the-art classification-based method on several tasks (prediction, motif discovery, variant scoring), in order to contextualize RBPNet's performance. However, these models are of very different nature and while both are modeling protein-RNA interaction, RBPNet aims to answer the question of "Assuming there is eCLIP signal, how does it distribute across nucleotides, given the observed sequence?" rather than "Is this sequence expected to generate a signal that passes the peak-calling threshold?". Nevertheless, our results demonstrate that identification of protein-RNA interactions at nucleotide-resolution can not be solved by current classification-based models, stressing the need for new approaches and highlighting the novelty of RBPNet.

Unspecific background signal, as well as experimental biases, is an inherent issue of CLIP-based protocols and thus downstream analysis, as bias towards certain sequence elements confounds the learning of genuine, protein-specific sequence features by modeling approaches. For classification-based models, few strategies have been developed to prevent models from learning bias instead of true signal. For instance, Pysster [6] compiles its negative labeled training set for a given RBP by sampling sequences from binding regions of other RBPs, thus explicitly introducing the same sequence biases of the positive set into the negative set, rendering them non-discriminative for the two classes. Similarly, DeepRiPe performs multi-task learning for several RBPs and only trains on sequences that harbor at least one experimentally identified binding site, such that biases associated with CLIP peaks are present in all input sequences. Yan et al. [67] address the RNase T1 cleavage bias towards guanine in the context of PAR-CLIP and HITS-CLIP by replacing nucleotides in a short window up-and downstream of the input sequence viewpoint with uniformly drawn random nucleotides. In contrast to the above strategies, which exclusively rely on manual manipulation of the training data, RBPNet accounts for experimental bias directly as part of its architecture by modeling the control signal as an auxiliary task. Specifically, RBPNet learns a component which explains the difference in signal shape of the experiment and a paired control, the target track, which is expected to be depleted of experimental bias and instead enriched in protein-specific signal. As the *target* signal is not observed experimentally, we instead quantitatively evaluated the ability of RBPNet tracks to recover known RBP-binding motifs and showed that the *target* track recovers motifs significantly better than the *total* track, with the later modeling the (possibly biased) observed signal. Orthogonal comparison with 5-mers derived from in vitro experiments further showed that the RBPNet *target* track performs comparable to PEKA, a state-of-the-art de novo motif discovery tool. This result is remarkable, as RBPNet is not a motif finder by design and the extracted motifs are derived only in retrospect via model interrogation. Both the RBPNet *target* and *total* tracks outperformed DeepRiPe on the task of in vitro motif recovery, which highlighted the advantage of learning directly on the bulk of raw experimental signal rather than a compressed representation in the form of binary labels, which may be associated with loss of information. The fact that RBPNet is trained on up to a million (in case of HNRNPC) regions enriched in CLIP-seq signal per RBP may further contribute towards its generalization power. In contrast, classifiers such as DeepRiPe [18] or Pysster [6] are trained on datasets the size of 10–100 thousand samples.

We demonstrated that RBPNet models can be utilized to score the impact of sequence variants on protein-RNA interaction via in silico mutagenesis. For instance, we showed that RBPNet scores single-nucleotide variant within splice junctions significantly higher than randomly selected background variants. In addition, analysis of non-coding transcribed regions revealed that RBPNet variant impact scores are higher in conserved regions, as measured by PhyloP score. This indicates that RBPNet models learned to associated individual nucleotides with the presence or absence of CLIP-seq peaks and that in silico probing of the RBPNet model can improve our mechanistic understanding of protein-RNA interaction.

While RBPNet was shown to predict the count distribution across an input sequence exceptionally well, the lack of a notion of absolute signal hinders a direct comparison of position-wise probabilities between different RNA sequences. This may impose limitation when comparing the binding affinity of two alleles in the context of variant scoring. By design, RBPNet predicts the nucleotide-wise distribution of CLIP-seq signal, rather than the absolute CLIP-seq signal, for a given sequence. Therefore, RBPNet can not natively predict which of two given alleles is expected to have higher CLIP-seq signal. RBPNet may still be utilized for variant scoring by observing that variants which result in gain or loss of CLIP-seq signal will likely alter the distribution of CLIP-seq signal. For instance, a sequence without binding affinity to the target RBP is (in the idealized case) expected to result in the prediction of a uniform CLIP-seq distribution by RBPNet, while a sequence containing a binding motif of the target RBP will result in prediction of a unimodal distribution with high probability mass at the expected crosslinking site. Disruption of the binding motif through a SNV will result in a drastic reallocation of probability mass, which one can quantify. Nevertheless, the lack of absolute binding affinity may hinder RBPNet from accurately ranking the alleles with respect to their binding binding to a target RBP and may explain the diminished performance on tasks such as allele-specific binding scoring, as seen in Fig. 5. Therefore, future work may explore the feasibility of absolute signal prediction, for instance by incorporating transcript abundance coefficients as model covariates. This is a complex task, as different RBPs encounter transcripts at different stages of their life-cycle. Therefore, different estimates of pre-or mature RNA abundance may be required, depending on prior knowledge of the protein at hand. A potential advantage of RBPNet variant scoring is that it can detect instances where a variant alters the *location* of protein-RNA interaction, while not affecting the overall binding affinity of the allele, as such variants induce measurable changes in the predicted CLIP-seq distribution. This indicates that predictions obtained from RBPNet and classifiers (such as DeepRiPe) may complement each other.

Although we demonstrated the power of our approach on several tasks, including CLIP signal shape imputation, identification of bona fide RBP-binding motifs and variant impact scoring, we believe that our results will pave the way for further downstream applications. A promising future application of RBPNet may be in silico peak calling for the translation of the predicted signal shape into binding sites-a task that is usually performed by peak calling algorithms in the context of experimental CLIP signal. While this task may require prediction of absolute CLIP-seq signal, many peak callers compute position-wise binding site thresholds on a transcript-to-transcript basis (Clipper[2]) or with respect to the local neighborhood (Clippy[3], Paraclu[4]), a strategy that may be adapted for RBPNet.

Avsec et al. [2] demonstrate that the BPNet model may be used to unravel the motif syntax underlying TF cooperativity. In a similar way we envision, as future direction, the application of RBPNet to discover the sequence rules of RBP cooperativity, something that has been only rarely addressed by previous studies. However, CLIP datasets vary greatly in quality, with respect to sequencing depth, number of replicates, replicate consistency, signal-to-noise ratio, and presence or absence of control libraries. While the unique feature of RBPNet to disentangle true signal from noise can in principle enable the accurate identification of composite binding motifs and RBP cooperativity while mitigating the effect of confounding sequence bias, a systematic evaluation of CLIP-seq dataset quality will be necessary to achieve this goal.

Conclusions

We presented RBPNet, a sequence-to-signal model that predict the distribution of crosslinking events across an input RNA sequence at single-nucleotide resolution. Training and evaluation of RBPNet on 103 eCLIP datasets showed high performance of RBPNet in terms of signal shape correlation, while evaluation on miCLIP and iCLIP datasets demonstrated the models generalization to other CLIP-based protocols. We utilized RBPNet's ability to handle variable-length input sequence to perform inference on whole-transcript and showed that predicted high-probability positions coincide with PureCLIP peaks, outperforming state-of-the-art classifiers. To account for experimental biases, we additionally modeled the signal distribution of paired control experiments and derived a de-biased component, the RBPNet *target* track, which is enriched in protein-specific signal. We showed that feature importance analysis of the de-biased RBPNet *target* yielded informative sub-sequences which recall in vitro motifs at levels comparable to state-of-the-art motif detectors. Finally, we demonstrated RBPNet's ability to score the impact of SNVs on protein-RNA interaction, which may enable prioritization disease-associated variants that disrupt regulatory RNA sequence by causing gain or disruption of RBP-binding sites. RBPNet represents a significant milestone towards full in silico imputation of protein-RNA interaction, while model interpretation suggest that learning on the raw CLIP signal captures more experimental variants, improving our mechanistic understand of protein-RNA interaction.

[2] https://github.com/YeoLab/clipper

[3] https://github.com/ulelab/clippy

[4] https://gitlab.com/mcfrith/paraclu

Methods

Data and preprocessing

ENCODE eCLIP

A total of 103 enhanced CLIP (eCLIP) datasets across 103 RBPs from the HepG2 cell line were obtained from the ENCODE database [62]. Each dataset consists of an eCLIP experiment with two replicates and one size-matched input (SMInput) control experiment, which omits the protein-specific immunoprecipitation step and is thus enriched in unspecific background signal. For each eCLIP and SMInput experiment, aligned R2 reads (i.e., reads whose start positions likely correspond to the position immediately downstream of the RBP cross-linking site) were extracted from the experiment BAM file via SAMtools [42]. Next, reads obtained from the both eCLIP replicates were merged and $5'$ read-start coverage for both plus and minus strands was computed via BEDtools [51].

miCLIP

m^6A individual-nucleotide resolution UV crosslinking and immunoprecipitation (miCLIP) datasets for HEK293T and mESC cells were obtained from Kortel et al. [37], comprising 4 and 2 replicates, respectively. In addition, miCLIP datasets for the mESC cell line are paired with 2 replicates of a METTL3 KO control experiment. For all datasets, bigWig files of crosslink count signals were directly obtained from the Gene Expression Omnibus (GEO) at the accession number GSE163500. Note that bigWig files of reads without C-to-T transitions were selected, as these reads represent read-through events and would result in unspecific truncation count signal. Lastly, replicates of each dataset were merged by summing of the position-wise crosslink counts.

iCLIP

Individual-nucleotide CLIP (iCLIP) datasets for TDP43 and PTBP1, each with two replicates and without control experiments, were obtained from Hallegger et al. [23] and Haberman et al. [20], respectively. Replicates were downloaded from the Sequence Read Archive (SRA) with accession codes ERS10930255 and ERS10930256 for TDP43 and ERR1588764 and ERR1588765 for PTBP1 and processed as described in [64]. The source code for the processing pipeline is available at https://github.com/ulelab/ncawareclip.

Selecting candidate sites for training

In order to speed up convergence of RBPNet, it is important to restrict model training to regions with significant crosslink count signal. In the context of BPNet [2], the authors therefore performed peak calling on ChIP-nexus data to select a set of regions highly enriched in count signal. However, recent work by Toneyan et al. [59] suggests that peak callers select sites too conservatively, which may result in under-fitting of sequence-to-signal models.

To train RBPNet on ENCODE eCLIP datasets, we select a large set of candidate sites as follows. Given the set of genes retrieved from the GENCODE (version 40) [14], a sliding window of size 100 is shifted over each gene ($stride = 1$), and the total number of counts within each window, as well as the highest positional count, is obtained. Next, a p-value is computed for each window via a Poisson test by comparing the observed window counts to the expected counts, given the gene-level crosslink counts and the gene-length. At each step, windows with a p-value < 0.01, a minimum window count of $N = 8$ and a minimum count *height* (i.e., maximum position-wise count within the window) of $H = 2$ are recorded as candidates and the sliding window is shifted forward by 50 nucleotides. This avoids clusters of redundant candidate sites within transcript regions. Finally, selected 100 *nt* windows are extended symmetrically upstrea and downstream to a final length of 300 *nt*. Note that since RNA is a stranded molecule, counts are obtained for each gene in a strand-specific manner.

For miCLIP datasets, similar parameters together with GENCODE vM23 for the mESC cell line where used. For iCLIP datasets, the minimum window threshold was reduced to $N = 4$, due to a lower sequencing depth.

RBPNet architecture

The body model architecture of RBPNet was inspired by BPNet [2]. RBPNet takes as input a 300-nt RNA sequence, which is one-hot encoded by mapping the bases A, C, G, and U to binary vectors [1, 0, 0, 0], [0, 1, 0, 0], [0, 0, 1, 0], and [0, 0, 0, 1], respectively. The 300×4 dimensional input is then fed into a 1D convolution layer with 128 filters of size 12, followed by 9 residual blocks. Each residual block consists of (1) a 1D dilated convolution layer with 128 filters of size 6 and exponentially increasing dilation factor, (2) a batch normalization layer, (3) a ReLU activation, and (4) a dropout layer with a dropout rate of 0.25, respectively. The output of the last residual block (hereafter referred to as "bottleneck" layer) then serves as input to one or more output heads, where each output head corresponds to one of the modeling tasks, for instance the prediction of eCLIP signal and (optionally) SMInput shape. This is outlined schematically in Fig. 1b. Each output head consist of a transposed 1D convolutional layer with a single filter of size 20, mapping the bottleneck feature map to a 300-dimensional output vector, which corresponds to the position-wise probabilities of the count distribution within the input window. Notably, as in BPNet [2], same-padding and no pooling is used across all convolution operations in order to conserve the one-to-one correspondence of input sequence positions, feature maps, and outputs.

RBPNet training

Prior to model training, candidate sites obtained in the "Selecting candidate sites for training" section were split chromosome-wise into validation (chr2, chr9, chr16), hold-out (chr1, chr8, chr15), and train (all other autosomes) sets for both human and mouse cell lines. RBPNet is trained using the Adam optimizer [35] and an initial learning rate (LR) of 0.004. Training is performed for a maximum of 50 epochs with an early-stopping criteria such that training terminates prematurely if the validation loss did not decrease within the last 10 epochs. In addition, a LR schedulers is used such that the LR is halved each time the validation loss did not improve within the last 6 epochs.

RBPNet loss

The output vector p^h_{pred} of each track h (e.g., eCLIP or SMInput) is used to parameterize a multinomial distribution of read-start counts. For a given training instance, the loss is then computed as the negative log-likelihood of the observed (true) counts c^h_{obs}, given the total counts n^h_{obs} in the input region and the probability vector p^h_{pred}. That is, the model's loss L^h on a task h is defined as

$$L^h = L(p^h_{pred},\ c^h_{obs},\ n^h_{obs}) = -\log p_{mult.}(c^h_{obs} \mid p^h_{pred},\ n^h_{obs}) \tag{1}$$

The total loss L is then obtained by taking the sum over all task-specific losses.

RBPNet bias correction

Experimental bias can lead to unspecific eCLIP signal, severely impacting the downstream binding preference analysis. Therefore, CLIP-seq experiments are usually paired with a control experiment to measure the abundance of background signal at each locus. Assuming that a single read-start count is observed either due to true protein-specific (*target*) signal or experimental bias (*control*), RBPNet models the total CLIP-seq signal as an additive multinomial mixture of the target and bias distributions. That is,

$$p_{total} = \pi \times p_{target} + (1 - \pi) \times p_{control}\ , \tag{2}$$

where p_{total} is the probability vector of the total (e.g., eCLIP) signal, while $p_{control}$ and p_{target} are the probability vectors of the control and (unobserved) target signal, respectively. Furthermore, $\pi/(1-\pi)$ is the relative intensity of the target over the bias signal, given by a mixing coefficient π. Note that π, p_{target} and $p_{control}$ are learned directly from sequence for each RBP. Note that we explicitly chose to mix target and control signals additively, rather than multiplicatively, as it allows for a clean separation of both tracks. Modeling of biases in the total track multiplicatively requires a non-zero (and possibly large) probability at the respective location in the target track, which may lead to bias artifacts in the target track. To ensure that $p_{control}$ properly approximates the *control* signal distribution over the input sequence, a combined loss on the *total* and *control* tracks is defined as

$$L = L_{CLIP}(p_{total},\ c_{CLIP},\ n_{CLIP}) + L_{Ctrl}(p_{control},\ c_{Ctrl},\ n_{Ctrl})\ , \tag{3}$$

such that $p_{control}$ is penalized to match the distribution of counts in the control experiment. Once the RBPNet model is trained we can obtain an approximation of the bias-free signal component as p_{target}. A graphical outline of a RBPNet forward pass with bias correction via an additive mixture of *target* and *control* signal is shown in Additional file 1: Fig. 1.

Estimating the additive mixing coefficient

The contribution of target and bias signal towards the total signal is expected to be dependent on the input sequence. For instance, in the presence of multiple RBP-binding motifs, the majority of counts may be observed due to protein-specific crosslinking, while under absence of clear binding motifs, crosslinking biases may dominate. RBPNet therefore estimates the multinomial mixing coefficient π from the input sequence. Given the feature map of the bottleneck layer (the "RBPNet architecture" section), filter-wise global average pooling is performed along the sequence axis. The resulting 128-dimensional representation of the sequence is then fed into a 1-unit dense layer with linear activation to predict the logit of the mixing coefficient.

Disentangling target and control signals

Given Equation 2 together with the total number of counts N, we can disentangle the eCLIP signal into the expect control and target counts:

$$E_{target} = \pi \times p_{target} \times N \tag{4}$$

$$E_{control} = (1 - \pi) \times p_{control} \times N \tag{5}$$

Sequence importance scores

To identify RNA sequence features that contributed significantly to the predicted signal distribution, we compute integrated gradients (IG) attribution scores [58] of input sequence with respect to the output probability vector p for each track. This way, we obtain separate attribution maps for predicted total, target and control signals.

By default, the IG attribution method assumes a classification-based setting, where gradients are computed with respect to the output probability of a target class of interest. For instance in the context of classification-based models, attributions may be computed with respect to a single output neuron describing the binding probability of the target RBP to the input RNA. Here, the resulting feature importance values quantify how much each feature contributed towards the target class. For instance, DeepRiPe employs IG to identify nucleotides that were contributed towards predicting an input sequence as "bound" for a target RBP. In contrast to classification-based methods, RBPNet predicts a 1D profile for each RBP and input sequence. Computing IG attribution maps with respect to only a single position in the output track may draw an incomplete picture of nucleotide-wise contributions towards the predicted signal footprint. We therefore introduce a generalization of IG from scalar to 1D profile outputs.

Given an observed input x and a baseline input x', the IG score of an input feature x_i is defined as

$$IG_i(x) := (x_i - x_i') \times \int_{\alpha=0}^{1} \frac{\partial F(x' + \alpha \times (x - x'))}{\partial x_i} d\alpha \tag{6}$$

where F is a scalar function. In the simplest case of binary classification, where the deep neural network f has a scalar output, $F = f$. In the case of multi-class classification, F is usually defined as $F(x) = \sum_i p_i y_i$, where $p = f(x)$ is the multi-class probability vector and y the true label vector with $y_i \in 0, 1$, such that IG scores of x are obtained with respect to its true class. A natural extension of F to count data is given by

$$F(x) = \sum_i p_i c_i \tag{7}$$

where p is the multinomial probability vector and c is the vector of true counts, with the desirable effect that predictions at positions with high counts will dominate the input gradients. The extension of F in (8) has two major drawbacks. First, it requires true counts c for a given sequence to compute attribution scores and second, it might up-weight positions with high counts that are due to experimental bias. We thus reformulate F as

$$F(x) = \sum_i p_i \times stop\_grad(p_i) \tag{8}$$

where $stop\_grad(x)$ stops gradients flow and treats x as a constant.

In other words, instead of computing gradients with respect to the scalar of a single output neuron, we compute gradients of the RNA sequence nucleotides with respect to the sum of the output profile, weighted by a constant version of itself. The weighting ensures that output positions with high probability contribute more towards the nucleotide-wise feature importance scores than low-probability positions.

Note that the proposed generalization of Integrated Gradients to output probability vectors is in analogy to Avsec et al.'s generalization of DeepLIFT [56] scores, described in [2].

By computing gradients with respect to p_{target} (rather than p_{total}), we explicitly remove contributions of the sequence towards experimental bias and thus focus solely nucleotides that contribute towards the protein-specific crosslinking signal. In general, attribution scores of the total, target and control tracks may be disentangled via

$$F_{total} = F_{target} + F_{control} = [\pi \times \sum_i c_i p_i^{target}] + [(1-\pi) \times \sum_i c_i p_i^{control}]. \tag{9}$$

Performance evaluation

Pearson correlation performance

Given the set of 300 nt sequences in the hold-out test set, Pearson correlation coefficients (PCC) between RBPNet predictions and the observed crosslink counts, merged between both replicates, were computed. For each eCLIP experiment, the final PCC performance metric is obtained by taking the mean PCC across all test-set sequences.

Comparison with PureCLIP crosslink sites

PureCLIP is a single-nucleotide peak caller that identifies significant crosslink (CL) sites by fitting a hidden Markov model over the CLIP count signal. To further validate profile predictions made by RBPNet, we investigated whether scores at positions within PureCLIP CL sites are significantly higher than scores outside of CL sites. To this end, we performed PureCLIP CL site "peak" calling on all ENCODE eCLIP and miCLIP experiments using default parameters. As suggested by Krakau et al. [38], replicate BAM files were merged to enable the use of signal information across all replicates during peak calling. For ENCODE eCLIP and the mESC miCLIP experiment, PureCLIP was additionally provided BAM files of the control experiment to refine the set of CL sites based on significant enrichment over the control. This was omitted for the HEK miCLIP experiment, as no paired control experiment was available.

For each dataset, transcripts on the hold-out chromosomes (the "RBPNet training" section) were intersected with PureCLIP CL sites, and only transcripts harboring at least one CL sites were retained, ensuring that the transcript was expressed in the given experiment. Next, whole-transcript predictions were performed with RBPNet, yielding a probability vector of CL enrichment summing up to 1 for each retained transcripts. To measure how well RBPNet predictions discriminate between CL and non-CL sites, the area under the ROC curve (auROC) and the average precision (AP) scores were computed for each transcript. The auROC score may be a more adequate measure of the discriminative power of RBPNet than AP, as the baseline of the AP score is subject to the imbalance of CL and non-CL sites, which are different for each transcript. Furthermore, the auROC is closely related to the Wilcoxon statistic and represents probability of ranking a randomly chosen CL sites above a randomly chosen non-CL site within each transcript, thus directly measuring the discriminative power of the model. Note that within-transcript evaluation is necessary because the position-wise RBPNet scores are subject to transcript length as well as the propensity of RBP binding within the transcript. The finally scalar performance metric for each experiment is then obtained by taking the mean auROC and AP scores across all transcripts.

We additionally evaluated RBPNet against DeepRiPe, a state-of-the-art deep learning model for prediction of protein-RNA interaction, on ENCODE eCLIP datasets. To this end, trained DeepRiPe models for 70 ENCODE HepG2 cell line were obtained from Ghanbari et al. [18]. As DeepRiPe is a classification-based model, the predicted binding probability score is assigned to an entire input regions. To make RBPNet and DeepRiPe comparable on the task of separate PureCLIP CL sites from non-CL sites, we obtained pseudo single-nucleotide resolution scores for DeepRiPe by applying same padding to the transcript sequence, before shifting a sliding window of 150 *nt* (DeepRiPe input size) across the sequence and assigning the prediction score to the center position of the current window.

Motif discovery and evaluation

RBPmap motif evaluation

To quantitatively evaluate the ability of RBPNet to recover known RBP-binding motifs in its sequence attribution maps, we compare high-attribution sub-sequences with known binding motifs in the form of position-weight matrices (PWMs) reported in literature. To this end, we first gathered the PWMs of 29 RBPs with both ENCODE eCLIP experiments and reported literature motifs from the RBPmap database [50]. Next, for each eCLIP experiment, the top 5000 ENCODE narrow peaks were selected, and profile predictions were performed on a 300-*nt* window around the 5′ end of the peak, as this position has previously been reported to harbor the CL site [10]. After computing attribution maps with respect to the RBPNet *total*, *target*, and *control* tracks, the 5-mer with highest sum of attribution (IG_{sum}) was extracted for each sequence and track. The similarity of each 5-mer to the reference PWM(s) of the RBP was then computed as the mean of the position-wise Jensen-Shannon divergence (JSD) to base 2, a symmetric version of the KL divergence within the bounds [0, 1], where 1 indicates perfect similarity. To account for cases where 5-mers represent truncated motifs or match the reference PWM at a different position offset (i.e., shifted upstream or downstream), we slide each 5-mer over its reference PWM, with a required overlap fraction of 3 *nt*. At each shift, the JSD is computed and the final similarity of the 5-mer, and the PWM is taken as the maximum similarity over all shift. In cases in which more than one motif PWM is reported for a given RBP in the RBPmap database, similarity computation is performed with respect to all PWMs and the final similarity score is taken as the maximum similarity between the 5-mer and all PWMs of that RBP.

In vitro motif evaluation

In vitro data on protein-RNA interaction was obtained in the form of k-mer z-scores for RNA-Bind-n-Seq (RBNS) and RNAcompete experiments from Dominquez et al. [10] and Ray et al. [53], respectively. For RBNS, 5mer enrichment scores (*R scores*) for 78 RBPs were obtained from the ENCODE resource, using accession numbers listed in [10]. For each RBP, the *R scores* for the concentration with the highest enrichment were converted to z-scores, by calculating their mean and standard deviation. RNAcompete 7-mer z-scores for 80 RBPs were obtained from Ray et al. [53]. In cases where both RNA-Bind-N-Seq and RNAcompete were available for a particular protein, we prioritized RBNS for downstream analysis, as RBNS z-scores were readily available for 5-mers, whereas RNAcompete required transformation from 7-mer to 5-mer scores. The conversion of 7-mer scores to 5-mer scores was performed by calculating the mean score across all 7-mers that contain a given 5-mer. 7-mers which contain a particular 5-mer more than once were considered as many times as the number of occurrences of the contained 5-mer. For illustration, when calculating the arithmetic mean of z-scores for a 5-mer "UUUUU," the 7-mer "UUUUUUG" would be considered twice ("[UUUUU]UG," "U[UUUUU]G"). In vitro 5-mers of each RBP were then sorted in a descending order based on their enrichment scores and ranked from most (ranked 1st) to least enriched (ranked last).

For evaluation of RBPNet with in vitro 5-mers, 5-mers with highest IG_{sum} were extracted from RBPNet attribution maps of the top 5000 ENCODE narrow peaks for each track, as described in the "RBPmap motif evaluation" section. Furthermore, DeepRiPe ENCODE models were obtained from [18], and unique 5-mer counts were obtained in a similar manner by first computing IG attribution maps on a 150-*nt* input window around ENCODE narrow peaks and subsequently selecting 5-mers of highest IG_{sum} for each narrow peak sequence (the "RBPmap motif evaluation" section). For each unique RBPNet and DeepRiPe 5-mer, a relevance score was then computed by taking the sum of IG_{sum} scores. RBPNet and DeepRiPe 5-mers were then sorted decreasingly with respect to their relevance score. Lastly, we obtained 5-mer enrichment scores calculated with PEKA[5] (v0.1.6), a motif discovery tool, for all ENCODE eCLIP datasets from [39]

[5] https://github.com/ulelab/peka

(reference: Additional File 5). We used PEKA-scores that were produced with Clippy peaks [64] to rank the k-mers from most to least enriched. As DeepRiPe models were only available for 70 out of 103 ENCODE HepG2 RBPs, and only 27 of those had orthogonal in vitro data available, the evaluation of recall was therefore restricted to those proteins. Out of 27 eCLIP datasets, 16 were compared to RBNS and 11 were compared to RNAcompete for recall analysis. Finally, an in vitro recall score was computed for each RBP and method by taking the proportion of top 20 5-mers from the corresponding RBNS or RNAcompete dataset that were recovered among the top 20 5-mers in eCLIP, as ranked by the RBPNet tracks, DeepRipe and PEKA.

Consensus motif construction

Representative consensus motifs for each eCLIP library were constructed as follows. Given the set of k-mers obtained in the "RBPmap motif evaluation" section, k-mers were first sorted by their IG_{sum} score in descending order. Iterating from the top of the list, the first k-mer is used to seed an initial motif alignment, with consecutive k-mers being aligned (without gaps) to the seed k-mer by sliding the given k-mer over the seed alignment and requiring a minimum overlap of 3 *nt*. If no alignment with at least 3 matches is found, the k-mer is considered non-alignable and is instead used to seed a new motif alignment. Consecutive k-mers are aligned to seed alignments in the order of their creation and, if no sufficient alignment is found, are used to seed further motifs alignments on-the-fly. Subsequently, consensus motifs are constructed for each alignment by computing the position-wise nucleotide frequencies within alignments. Consensus motifs can then be prioritized based on the number of supporting k-mers in the underlying alignment. The motif finding procedure was implemented as part of the following Git repository:https://github.com/mhorlacher/metamotif.

Comparison of RBPNet motifs with existing databases

Motifs were obtained in form of position-weight matrices from the ATtRACT [19], RBPDB [8], mCross [11], and oRNAment [4] databases via RSAT [54] and additionally supplemented with motifs from the RBPmap [50] database. All database motifs were converted to the TRANSFAC format via RSAT convert-matrix. For each eCLIP experiment, the top-2 RBPNet motifs with respect to 5-mer support were selected and subsequently compared to all motifs of the corresponding RBP in each database. As a measure of similarity between two PWMs, the normalized correlation relative to the smallest of the two aligned matrices (NcorS) was computed using the RSAT "compare-matrices" command. As databases commonly report several motifs for a given RBP, the maximum similarity score was taken as the final similarity score between an RBPNet motif and motifs of the corresponding RBP in a database.

Variant impact scoring

The predicted distribution of counts by RBPNet is solely driven by the input RNA sequence, and thus one expects that single-nucleotide variants (SNVs) that fall within crucial sequence feature, such as binding motifs, will have a profound impact on the predicted signal footprint. Therefore, to approximate the impact of a SNV on RBP binding, we quantify the change of the distribution of counts of the alternative allele compared to the reference. To this end, we define the change of count distribution with respect to a given SNV as the KL divergence of the prediction on the SNV-associated allele from the prediction on the reference allele. This impact score is thus defined as

$$Impact_{KLD}(SNV) = KLD(p^{REF}, p^{ALT}) . \tag{10}$$

Scoring of gnomAD variants

GnomAD variants (v2.1, hg38 assembly) with PASS filter status were further filtered in order to keep single-nucleotide variants with available allele frequency, dropping singletons ($AC = 1$). They were then intersected with protein-coding genomic annotations (GENCODE v40) and filtered for mutations within 3′UTR or 5′UTR or intronic within 200 nt of a splice site. This resulted in a set of 1570,195 mutations. As a proxy of the negative selection associated to these genomic positions, the mutations were grouped based on their allele frequency, from common mutations ($AF >= 0.05$) to rare mutations ($AF < 0.001$). In addition, they were annotated with the PhyloP 100w score, measuring evolutionary conservation across 100 vertebrates, and grouped so as to separate highly constrained genomic positions (positive PhyloP scores) from neutrally or fast evolving positions (negative PhyloP scores). Finally, a representative set of set of 15 RBPNet models was selected to score the putative impact mutations on RBP binding. RBPs were selected based on genomic region preference and RNA processing functions (obtained from [63]). For instance, selected RBPs are involved in translation regulation (e.g., NCPB2), splicing (e.g., RBFOX2), or post-transcriptional regulation (e.g., SND1). For each RBP, the average and standard error of the impact score was measured for each bin of sequence constraint, evaluated either through the allele frequency or the PhyloP score. Finally, variants were evaluated w.r.t. a positive relationship between the predicted RBPNet impact score and the degree of sequence constraint.

Scoring of allele-specific binding (ASB) events

Allele-specific binding variants were obtained from Yang et al. [66] and filtered by removing variants with less than 20 reads across both alleles, in order to enrich for a robust set of ASB variants for downstream evaluation. Variants were further filtered by removing all variants with a neighboring variant in a 300-nt. This was done in order to remove potential false-positives (albeit at the expense removing true positives), as variants in close proximity create ambiguity with respect to the causal variant for ASB. RBPs with less than 10 ASB variants were not considered for analysis, resulting in a evaluation set of 44 RBPs. Finally, 10 random positions within the gene of each ASB variant were samples to generate a set of background, non-ASB variants.

Variant impact scoring of splicing mutations

Forty out of 103 RBPs with trained RBPNet-eCLIP models were manually annotated as related to splicing, following the annotations from Nostrand et al. [63] and the HGNC spliceosomal complex groups from the HGNC database [60]. Of these, 21 were further annotated as directly involved in the spliceosome. Next, a set of 260 experimentally validated splicing-related mutations was obtained from MutSpliceDB [47]. After excluding mutations with a distance of more than 10 nt from splicing junctions (defined as the first two and last two positions from intronic regions in human coding genes of GENCODE V40 [14]), a set of 232 mutations was retained. For negative controls, we retrieved 6087 mutations from gnomAD v2.1.1 ([31]) located within 100 nt upstream or downstream of the retained splicing mutations. Control mutations which intersected with the set of splicing-associated mutations were filtered out. Subsequently, RBPNet impact scoring (the "Variant impact scoring" section) was performed on all mutations. For each RBP, a one-sided Wilcoxon ranked sum test was performed to evaluate the enrichment of high-impact splicing-associated mutations over control mutations. P-values were corrected for multiple testing via Benjamini-Hochberg correction, and significance was tested for $\alpha = 0.05$. The same procedure was applied for 30 of the 70 DeepRiPe models found in common with the 40 RBPNet models, taking the models trained from ENCODE HepG2 using both sequence and genomic annotations. Here, the impact score was calculated as the absolute difference in prediction score for a given RBP between the alternative and the reference allele.

Supplementary Information

The online version contains supplementary material available at https://doi.org/10.1186/s13059-023-03015-7.

Additional file 1. Includes additional text on the expected variance of replicates, additional figures for the RBPNet architecture, consensus motifs, mixing coefficient analysis and variant scoring analysis.

Additional file 2. Includes Fig. S2, a collection of RBPNet-derived consensus motifs and their similarity to motifs of other databases.

Additional file 3. Review history.

Acknowledgements
Not applicable.

Peer review information
Kevin Pang was the primary editor of this article and managed its editorial process and peer review in collaboration with the rest of the editorial team.

Review history
The review history is available as Additional file 3.

Authors' contributions
M.H. and A.M. conceived the project. M.H. collected and processed the datasets and conceptualized and implemented the RBPNet model with help from O.W.; N.W. and L.M. performed the analysis of allele-specific binding events and splice-site mutations, respectively, with help from M.H.; K.K. performed the binding motif recall analysis on in vitro binding data. N.G. helped in processing datasets and training RBPNet models. O.W. and A.M. supervised and guided the project, with inputs from M.S., J.U., and J.G.; M.H. wrote the manuscript with help from K.K., N.W., A.M., and L.M. and inputs from O.W., J.G., and J.U.; all authors reviewed and approved the final manuscript.

Funding
Open Access funding enabled and organized by Projekt DEAL. This work was supported by the Helmholtz Association under the joint research school "Munich School for Data Science (MUDS)" to M.H., N.W., J.G. and A.M., the Deutsche Forschungsgemeinschaft (SFB/TR501 84 TP C01) to A.M. and L.M. and (SFB/Transregio TRR267) to J.G.; O.W.'s work was funded in part by the Novo Nordisk Foundation through the Center for Basic Machine Learning Research in Life Science (NNF20OC0062606). O.W. further acknowledges support from the Pioneer Centre for AI, DNRF grant number P1; K.K.'s and J.U.'s work was funded by the European Union's Horizon 2020 research and innovation program (835300-RNPdynamics). K.K. and J.U. further acknowledge support from The Francis Crick Institute, which receives its core funding from Cancer Research UK (FC001110), the UK Medical Research Council (FC001110), and the Wellcome Trust (FC001110).

Availability of data and materials
Code for RBPNet training, evaluation, feature importance analysis, and variant impact scoring is available at GitHub (https://github.com/mhorlacher/rbpnet) [28] and Zenodo (https://doi.org/10.5281/zenodo.8125355) [26]. Code for consensus motif construction is available at https://github.com/mhorlacher/metamotif. Code in both GitHub repositories is available under the MIT license.
All data processed in this study was obtained exclusively from public sources. CLIP-seq data was obtained from ENCODE [62] (eCLIP), Kortel et al. [37] (miCLIP), and Hallegger et al. [23] and Haberman et al. [20] (iCLIP). In vitro data on

protein-RNA interaction was obtained from Dominquez et al. [10] (RNA-Bind-n-Seq) and Ray et al. [53] (RNAcompete). Splicing-related mutations were obtained from MutSpliceDB [47], while further mutations were obtained from gnomAD [31]. Information on allele-specific binding events was taken from Yang et al. [66].

Declarations

Ethics approval and consent to participate
Not applicable.

Consent for publication
Not applicable.

Competing interests
The authors declare no competing interests.

Received: 23 September 2022 Accepted: 17 July 2023
Published online: 04 August 2023

References

1. Alipanahi B, Delong A, Weirauch MT, Frey BJ. Predicting the sequence specificities of DNA-and RNA-binding proteins by deep learning. Nat Biotechnol. 2015;33(8):831–8.
2. Avsec Ž, Weilert M, Shrikumar A, Krueger S, Alexandari A, Dalal K, et al. Base-resolution models of transcription-factor binding reveal soft motif syntax. Nat Genet. 2021;53(3):354–66.
3. Bailey TL, Boden M, Buske FA, Frith M, Grant CE, Clementi L, et al. MEME SUITE: tools for motif discovery and searching. Nucleic Acids Res. 2009;37(suppl_2):W202–8.
4. Benoit Bouvrette LP, Bovaird S, Blanchette M, Lécuyer E. oRNAment: a database of putative RNA binding protein target sites in the transcriptomes of model species. Nucleic Acids Res. 2020;48(D1):D166–73.
5. Bergstrand S, OBrien EM, Coucoravas C, Hrossova D, Peirasmaki D, Schmidli S, et al. Small Cajal body-associated RNA 2 (scaRNA2) regulates DNA repair pathway choice by inhibiting DNA-PK. Nat Commun. 2022;13(1):1–18.
6. Budach S, Marsico A. Pysster: classification of biological sequences by learning sequence and structure motifs with convolutional neural networks. Bioinformatics. 2018;34(17):3035–7.
7. Chen X, Liu Y, Xu C, Ba L, Liu Z, Li X, et al. QKI is a critical pre-mRNA alternative splicing regulator of cardiac myofibrillogenesis and contractile function. Nat Commun. 2021;12(1):1–18.
8. Cook KB, Kazan H, Zuberi K, Morris Q, Hughes TR. RBPDB: a database of RNA-binding specificities. Nucleic Acids Res. 2010;39(suppl_1):D301–8.
9. De Conti L, Baralle M, Buratti E. Neurodegeneration and RNA-binding proteins. WIREs RNA. 2017;8(2):e1394. https://doi.org/10.1002/wrna.1394.
10. Dominguez D, Freese P, Alexis MS, Su A, Hochman M, Palden T, et al. Sequence, structure, and context preferences of human RNA binding proteins. Molecular cell. 2018;70(5):854–67.
11. Feng H, Bao S, Rahman MA, Weyn-Vanhentenryck SM, Khan A, Wong J, et al. Modeling RNA-binding protein specificity in vivo by precisely registering protein-RNA crosslink sites. Mol Cell. 2019;74(6):1189–204.
12. Findlay SD, Romo L, Burge CB. Quantifying negative selection in human 3' UTRs uncovers constrained targets of RNA-binding proteins. bioRxiv. 2022;2022–11.
13. Flynn RA, Belk JA, Qi Y, Yasumoto Y, Wei J, Alfajaro MM, et al. Discovery and functional interrogation of SARS-CoV-2 RNA-host protein interactions. Cell. 2021;184(9):2394–411.
14. Frankish A, Diekhans M, Jungreis I, Lagarde J, Loveland JE, Mudge JM, et al. GENCODE 2021. Nucleic Acids Res. 2021;49(D1):D916–23.
15. Fredericks AM, Cygan KJ, Brown BA, Fairbrother WG. RNA-binding proteins: splicing factors and disease. Biomolecules. 2015;5(2):893–909. https://doi.org/10.3390/biom5020893. www.ncbi.nlm.nih.gov/pmc/articles/PMC4496701/.
16. Garcia-Moreno M, Järvelin AI, Castello A. Unconventional RNA-binding proteins step into the virus-host battlefront. Wiley Interdiscip Rev RNA. 2018;9(6):e1498.
17. Gebauer F, Schwarzl T, Valcárcel J, Hentze MW. RNA-binding proteins in human genetic disease. Nat Rev Genet. 2021;22(3):185–98.
18. Ghanbari M, Ohler U. Deep neural networks for interpreting RNA-binding protein target preferences. Genome Res. 2020;30(2):214–26.
19. Giudice G, Sánchez-Cabo F, Torroja C, Lara-Pezzi E. Attracta database of RNA-binding proteins and associated motifs. Database. 2016;2016.
20. Haberman N, Huppertz I, Attig J, König J, Wang Z, Hauer C, et al. Insights into the design and interpretation of iCLIP experiments. Genome Biol. 2017;18(1):1–21.
21. Hafner M, Landthaler M, Burger L, Khorshid M, Hausser J, Berninger P, et al. Transcriptome-wide identification of RNA-binding protein and microRNA target sites by PAR-CLIP. Cell. 2010;141(1):129–41.
22. Hafner M, Katsantoni M, Köster T, Marks J, Mukherjee J, Staiger D, et al. CLIP and complementary methods. Nat Rev Methods Prim. 2021;1(1):1–23.
23. Hallegger M, Chakrabarti AM, Lee FC, Lee BL, Amalietti AG, Odeh HM, et al. TDP-43 condensation properties specify its RNA-binding and regulatory repertoire. Cell. 2021;184(18):4680–96.
24. 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 research. 2017;45(19):11004–18.

25. Hentze MW, Castello A, Schwarzl T, Preiss T. A brave new world of RNA-binding proteins. Nat Rev Mol Cell Biol. 2018;19(5):327–41.
26. Horlacher M, Wagner N, Moyon L, Kuret K, Goedert N, Salvatore M, et al. Zenodo. 2023. https://doi.org/10.5281/zenodo.8125355.
27. Horlacher M, Oleshko S, Hu Y, Ghanbari M, Vergara EE, Mueller N, et al. A computational map of the human-SARS-CoV-2 protein–RNA interactome predicted at single-nucleotide resolution. NAR Genomics and Bioinformatics. 2023;5(1):lqad010.
28. Horlacher M, Wagner N, Moyon L, Kuret K, Goedert N, Salvatore M, et al. GitHub. 2023. https://github.com/mhorlacher/RBPNet. Accessed 28 Sept 2022.
29. Huppertz I, Attig J, D'Ambrogio A, Easton LE, Sibley CR, Sugimoto Y, et al. iCLIP: protein–RNA interactions at nucleotide resolution. Methods. 2014;65(3):274–87.
30. Izumikawa K, Nobe Y, Ishikawa H, Yamauchi Y, Taoka M, Sato K, et al. TDP-43 regulates site-specific 2-O-methylation of U1 and U2 snRNAs via controlling the Cajal body localization of a subset of C/D scaRNAs. Nucleic Acids Res. 2019;47(5):2487–505.
31. Karczewski KJ, Francioli LC, Tiao G, Cummings BB, Alföldi J, Wang Q, et al. The mutational constraint spectrum quantified from variation in 141,456 humans. Nature. 2020;581(7809):434–43.
32. Kazan H, Ray D, Chan ET, Hughes TR, Morris Q. RNAcontext: a new method for learning the sequence and structure binding preferences of RNA-binding proteins. PLoS Comput Biol. 2010;6(7):e1000832.
33. Ke S, Pandya-Jones A, Saito Y, Fak JJ, Vågbø CB, Geula S, et al. m6A mRNA modifications are deposited in nascent pre-mRNA and are not required for splicing but do specify cytoplasmic turnover. Genes Dev. 2017;31(10):990–1006.
34. Kelley DR, Reshef YA, Bileschi M, Belanger D, McLean CY, Snoek J. Sequential regulatory activity prediction across chromosomes with convolutional neural networks. Genome Res. 2018;28(5):739–50.
35. Kingma DP, Ba J. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980. 2014.
36. König J, Zarnack K, Rot G, Curk T, Kayikci M, Zupan B, et al. iCLIP reveals the function of hnRNP particles in splicing at individual nucleotide resolution. Nat Struct Mol Biol. 2010;17(7):909–15.
37. Körtel N, Rücklé C, Zhou Y, Busch A, Hoch-Kraft P, Sutandy FR, et al. Deep and accurate detection of m6A RNA modifications using miCLIP2 and m6Aboost machine learning. Nucleic Acids Res. 2021;49(16):e92–e92.
38. Krakau S, Richard H, Marsico A. PureCLIP: capturing target-specific protein-RNA interaction footprints from single-nucleotide CLIP-seq data. Genome Biol. 2017;18(1):1–17.
39. Kuret K, Amalietti AG, Jones DM, Capitanchik C, Ule J. Positional motif analysis reveals the extent of specificity of protein-RNA interactions observed by CLIP. Genome Biol. 2022;23(1):1–34.
40. Labeau A, Fery-Simonian L, Lefevre-Utile A, Pourcelot M, Bonnet-Madin L, Soumelis V, et al. Characterization and functional interrogation of the SARS-CoV-2 RNA interactome. Cell Rep. 2022;39(4):110744.
41. Lambert N, Robertson A, Jangi M, McGeary S, Sharp PA, Burge CB. RNA Bind-n-Seq: quantitative assessment of the sequence and structural binding specificity of RNA binding proteins. Mol Cell. 2014;54(5):887–900.
42. Li H, Handsaker B, Wysoker A, Fennell T, Ruan J, Homer N, et al. The sequence alignment/map format and SAMtools. Bioinformatics. 2009;25(16):2078–9.
43. Linder B, Grozhik AV, Olarerin-George AO, Meydan C, Mason CE, Jaffrey SR. Single-nucleotide-resolution mapping of m6A and m6Am throughout the transcriptome. Nat Methods. 2015;12(8):767–72.
44. Maticzka D, Lange SJ, Costa F, Backofen R. GraphProt: modeling binding preferences of RNA-binding proteins. Genome Biol. 2014;15(1):1–18.
45. Meyer KD. DART-seq: an antibody-free method for global m6A detection. Nat Methods. 2019;16(12):1275–80.
46. Molleston JM, Cherry S. Attacked from all sides: RNA decay in antiviral defense. Viruses. 2017;9(1):2.
47. Palmisano A, Vural S, Zhao Y, Sonkin D. MutSpliceDB: a database of splice sites variants with RNA-seq based evidence on effects on splicing. Hum Mutat. 2021;42(4):342–5. https://doi.org/10.1002/humu.24185, _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/humu.24185. Accessed 28 Sept 2022.
48. Pan X, Rijnbeek P, Yan J, Shen HB. Prediction of RNA-protein sequence and structure binding preferences using deep convolutional and recurrent neural networks. BMC Genomics. 2018;19(1):1–11.
49. Park CY, Zhou J, Wong AK, Chen KM, Theesfeld CL, Darnell RB, et al. Genome-wide landscape of RNA-binding protein target site dysregulation reveals a major impact on psychiatric disorder risk. Nat Genet. 2021;53(2):166–73.
50. Paz I, Kosti I, Ares M Jr, Cline M, Mandel-Gutfreund Y. RBPmap: a web server for mapping binding sites of RNA-binding proteins. Nucleic Acids Res. 2014;42(W1):W361–7.
51. Quinlan AR, Hall IM. BEDTools: a flexible suite of utilities for comparing genomic features. Bioinformatics. 2010;26(6):841–2.
52. Ray D, Kazan H, Chan ET, Castillo LP, Chaudhry S, Talukder S, et al. Rapid and systematic analysis of the RNA recognition specificities of RNA-binding proteins. Nat Biotechnol. 2009;27(7):667–70.
53. Ray D, Kazan H, Cook KB, Weirauch MT, Najafabadi HS, Li X, et al. A compendium of RNA-binding motifs for decoding gene regulation. Nature. 2013;499(7457):172–7.
54. Santana-Garcia W, Castro-Mondragon JA, Padilla-Gálvez M, Nguyen NTT, Elizondo-Salas A, Ksouri N, et al. RSAT 2022: regulatory sequence analysis tools. Nucleic Acids Res. 2022;50(W1):W670–6.
55. Schmidt N, Lareau CA, Keshishian H, Ganskih S, Schneider C, Hennig T, et al. The SARS-CoV-2 RNA-protein interactome in infected human cells. Nat Microbiol. 2021;6(3):339–53.
56. Shrikumar A, Greenside P, Kundaje A. Learning important features through propagating activation differences. In: International conference on machine learning. PMLR; 2017. p. 3145–3153.
57. Sugimoto Y, König J, Hussain S, Zupan B, Curk T, Frye M, et al. Analysis of CLIP and iCLIP methods for nucleotide-resolution studies of protein-RNA interactions. Genome Biol. 2012;13(8):1–13.
58. Sundararajan M, Taly A, Yan Q. Axiomatic attribution for deep networks. In: International Conference on Machine Learning. PMLR; 2017. p. 3319–3328.
59. Toneyan S, Tang Z, Koo PK. Evaluating deep learning for predicting epigenomic profiles. Nature machine intelligence. 2022;4(12):1088–100.

60. Tweedie S, Braschi B, Gray K, Jones TEM, Seal R, Yates B, et al. Genenames.org: the HGNC and VGNC resources in 2021. Nucleic Acids Res. 2021;49(D1):D939–46. https://doi.org/10.1093/nar/gkaa980.
61. Van Nostrand EL, Pratt GA, Shishkin AA, Gelboin-Burkhart C, Fang MY, Sundararaman B, et al. Robust transcriptome-wide discovery of RNA-binding protein binding sites with enhanced CLIP (eCLIP). Nat Methods. 2016;13(6):508–14.
62. Van Nostrand EL, Freese P, Pratt GA, Wang X, Wei X, Xiao R, et al. A large-scale binding and functional map of human RNA-binding proteins. Nature. 2020;583(7818):711–9.
63. Van Nostrand EL, Pratt GA, Yee BA, Wheeler EC, Blue SM, Mueller J, et al. Principles of RNA processing from analysis of enhanced CLIP maps for 150 RNA binding proteins. Genome Biol. 2020;21(1):90. https://doi.org/10.1186/s13059-020-01982-9.
64. Varier RA, Sideri T, Capitanchik C, Manova Z, Calvani E, Rossi A, et al. m6A reader Pho92 is recruited co-transcriptionally and couples translation efficacy to mRNA decay to promote meiotic fitness in yeast. Elife. 2022;11(2022):e84034.
65. Wheeler EC, Van Nostrand EL, Yeo GW. Advances and challenges in the detection of transcriptome-wide protein-RNA interactions. Wiley Interdiscip Rev RNA. 2018;9(1):e1436.
66. Yang EW, Bahn JH, Hsiao EYH, Tan BX, Sun Y, Fu T, et al. Allele-specific binding of RNA-binding proteins reveals functional genetic variants in the RNA. Nat Commun. 2019;10(1):1–15.
67. Yan Z, Hamilton WL, Blanchette M. Graph neural representational learning of RNA secondary structures for predicting RNA-protein interactions. Bioinformatics. 2020;36(Supplement_1):i276–84.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Part V

DISCUSSION AND OUTLOOK

RNA-binding proteins have emerged as key actors of post-transcriptional gene regulation and are implicated in an abundance of human diseases [45, 183]. At the same time, they are at the center of a new generation of RNA-based therapeutics, such as Antisense Oligonucleotides (ASOs), which can modulate RNA processing by blocking cis-regulatory motifs. For instance, by obstructing splicing regulatory cis-elements, ASOs may change splice patterns and correct malign aberrant splicing [184]. It is clear that the design of therapeutic ribonucleotide sequences requires confident knowledge of the cis-regulatory landscape of the target RNA as well as its trans-acting factors. Here, sufficiently capable computational models may uncover critical factors for a given post-transcriptional process, identify their binding sites and predict the effects of their perturbation, thereby enabling *in silico* drug design. This thesis, together with its associated publications, contributes towards this goal. First, it successfully demonstrates the use of computational RBP-binding models for the discovery of potential host factors in viral infection by predicting several human RBPs to interact with the SARS-CoV-2 RNA [1]. Second, it surveys methods for the prediction of protein-RNA interactions through a comprehensive benchmark study, identifying state-of-the-art approaches and providing guidance for the development of high-performing predictive models [2]. Third, it proposes a novel deep learning method for directly modeling the raw signal of CLIP assays, enabling the prediction of protein-RNA interactions at unprecedented single-nucleotide resolution [3]. In the following, the associated publications, in particular their potential limitations as well as opportunities for further research, are discussed.

10.1 IMPROVING HUMAN-VIRUS PROTEIN–RNA INTERACTION PREDICTION

In core publication [1], a computational map of human-SARS-CoV-2 protein-RNA interactions is constructed by leveraging pysster [104] and DeepRiPe [153]. As these model are classification based and accept fixed-length sequence windows of 400nt and 100nt, respectively, the viral RNA was scanned via a sliding window approach and predictions were assigned to the center nucleotide. Besides incurring significant computational costs, this approach merely yields a predictions at pseudo-nucleotide resolution, as the predicted RBP binding site is not guaranteed to lie at the center of the classifier's input window. To address this, future investigation of human-virus protein-RNA interactomes may instead leverage RBPNet's ability to predict on variable-length sequences, allowing for inference of RBP-binding on the whole viral RNA at true nucleotide resolution in a single forward-pass. Further, while the identification of interactions between human proteins and viral RNAs may reveal important host factors, it is the active expression of these RBPs that ultimately dictates the permissiveness of individual tissues to viral infection. Future studies may therefore need take into account the differential expression of RBP between permissive and non-permissive tissues to identify crucial RBP host factors. Lastly, the study investigated SNVs in several viral variants of concern to identify whether a given SNV is associated with gain-or loss of RBP-binding, which could explain the increased virulence of a given viral variant. This is done by scoring the impact of the given SNV on binding of over hundred RBPs. As an alternative approach, one may first predicting the impact of variants on higher-level properties, such as RNA stability. Subsequently, RBPs involved in these processes may be further assess by investigating whether their binding is disrupted by the SNV.

10.2 Expanding the Benchmark of RBP-binding Prediction Methods

In the core publication [2], a large-scale benchmark study to evaluate protein-RNA interaction prediction methods is performed. While the study compiles and reviews 37 methods, a subset of 11 representative methods were sampled for benchmarking purposes due to computational constrains. Therefore, a natural extension of the benchmark is the incorporation of a greater number of methods in order to make more robust statements on the advantages or disadvantages of specific neural network architectures, such as LSTM networks or CNNs. Future version of the study's GitHub repository[1] may also include the release of precise instructions for method authors to add their own predictive models, in order to promote a continuous increase in benchmarked methods. A core motivation for constructing a large-scale benchmark of RBP-binding prediction methods was the identification of important model design choices shared among state-of-the-art methods. Consequently, future research may leverage the insights gained from this benchmark to develop a novel method which further improves on the current state-of-the-art. This may include the addition of input modalities beyond the primary RNA sequence, such as sequence conservation scores, intron/exon annotations and RNA secondary structure. An aspect not addressed in the benchmark study was the effect of peak calling on model performance. Future research may therefore perform sample generation with different peak callers (Section 3.2.3) prior to model training and evaluation. Further, this benchmark evaluates methods solely with respect to their classification performance. In practice, researchers may additionally be interested in aspects such as a method's runtime, its ability to identify RBP-binding motifs or how accurately it can score the impact of SNVs on RBP-binding. Future iterations of the benchmark framework therefore need to accommodate these additional tasks. Lastly, as the current benchmarking framework is solely concerned with classification-based approaches, new evaluation methods need to be devised to properly evaluate sequence-to-sequence approaches such as RBPNet [3].

10.3 Limitations and Future Directions of RBPNet

RBPNet [3] models protein-RNA interaction at single-nucleotide resolution. A notable limitation of the RBPNet model is the lack of *absolute* signal predictions. Classification-based methods, such as Pysster [104] or DeepRiPe [153], predict the probability that a given RNA sequence is bound by the protein of interest. In contrast, RBPNet predicts the *distribution* of protein-RNA crosslink counts such that the predicted probability mass at a certain nucleotide depends on the sequence context. For instance, if an RNA sequence contains a single RBP binding site, the mass of the predicted probability vector is expected to be centered at that site. On the other hand, sequences with multiple binding sites will show a spread of probability mass across all sites such that sites will have a lower mass on average, irrespectively of their binding affinity to the target protein. Fortunately, RBPNet enables predictions across whole transcripts, such that one may assess the RBP-binding affinity of a certain site relative to the rest of the transcript. Nonetheless, with regard to the comparison of the absolute binding affinity of two sites that reside within different transcripts, RBPNet shares the same limitations as current CLIP assays. As the number of observed crosslink counts depends on the abundance of

1 https://github.com/mhorlacher/Benchmark-RBP

the RNA, CLIP can not determine which of two sites corresponds to a higher stoichiometric binding affinity [72]. As the prediction of absolute crosslink counts necessitates the estimation of the RNA's expression, future research may explore the viability of using RNA abundance data measured by RNA-seq in the same cell lines, which is available in the ENCODE [71] database, as additional covariates for the RBPNet model. Recently, Linder et al. presented Borzoi, which enables accurate prediction of RNA-seq coverage from DNA sequence [180]. Extending Borzoi to directly predict CLIP coverage, either via fine-tuning or via additional prediction heads, may offer a promising avenue to model the absolute signal generated by CLIP experiments solely from sequence.

RBPNet aims to take a step towards *in silico* imputation of CLIP signal on novel RNA sequences. However in some cases, categorizing RNA sequences into discrete *bound* and *unbound* regions may be desirable. To this end, future research may explores *in silico* peak calling by applying peak callers to the output of RBPNet. This approach would necessitate the use of a null model to support statistical claims regarding the significance of peaks. Such a null model could be constructed by training RBPNet on data with shuffled crosslink counts within samples, which would preserve the local sequence bias while eliminating nucleotide-wise count information.

Future versions of RBPNet may incorporate improvements to the model's architecture. Recent research has shown that adding transformer layers on top of convolutional layers enhances model performance by capturing long-range interactions in regulatory genomics prediction tasks [149]. A similar design could enhance RBPNet's performance by facilitating the learning of interactions between spaced cis-regulatory binding motifs. However, a crucial limitation of common transformer layers is the quadratic time complexity of the self-attention operation. This would render predictions on whole transcript sequences intractable, as the size of pre-mRNAs sequences may be in the order of hundreds of kilo-bases. This problem may be partially addressed via spatial down-sampling of the feature maps, but would decrease the model's resolution. Future research may also consider the adoption of efficient transformer architectures, like the sliding-window attention approach employed in Longformer [185], to circumvent the trade-offs between input length and model resolution. As shown in [2], additional input modalities can greatly boost the performance of RBP-binding prediction methods. To increase the performance of RBPNet, modalities such as predicted or *in vivo* measured RNA secondary structures may be incorporated in future work. However, this may require a thorough investigation of secondary structure encoding strategies, such as the use of discrete structure alphabets [133] or the encoding of base-paring probabilities as a graph [186].

Benchmarking of several deep learning methods for RBP-binding prediction further showed that multi-label methods tend to outperform binary-classification methods [2]. To boost performance, a similar approach could be envisioned in a future version of RBPNet. Rather than training a single model for each CLIP dataset, a multi-task model which shares RBPNet's model body but contains a separate output head for each CLIP assay could be trained instead. Such a model might be considerably more computationally and statistically efficient, as several RBPs show similar binding preferences, thus promoting parameter-sharing between prediction tasks in the model's convolutional body. However, increasing the number of convolutional kernels may be necessary to capture a greater variety of unique concepts, including different binding motifs and motif interactions. Since post-transcriptional processes are primarily driven by RNA-RBP interactions, a model trained to predict binding for a large number of proteins may serve as a foundational model, which could subsequently be fine-tuned to predict other RNA properties, including half-life, localization, or translational efficiency. Lastly, multi-task modeling of CLIP data could offer new opportunities for enhanced bias correction. In the current version of RBPNet, the CLIP signal of an RBP is represented by a mixture of the predicted protein-specific signal and the control signal. In a multi-task setting, the CLIP signal may be modeled as a mixture of the protein-specific signal and the average signal from all other CLIP tracks. In essence, the binding specificity of a target protein is described as the deviation from the average specificity of all other proteins. This approach allows for the correction of biases that may be introduced during immunoprecipitation — a step omitted in the current control experiments of eCLIP [14].

Finally, to make RBPNet available to a wider audience of RNA researchers, RBPNet training and prediction workflows may be integrated into `flow.bio` [187], a bioinformatics workflow management system on the basis of NextFlow [188]. Trained models may also be deposited into Kipoi [189], a repository for machine learning methods in genomics.

10.4 OPEN CHALLENGES IN THE FIELD OF RBP-BINDING PREDICTION

Uncovering predictive sub-sequences through interpretation of machine learning methods trained to predict RBP-binding is crucial for uncovering an RBP's binding preferences and the regulatory code of post-transcriptional processes. Yet, existing approaches to summarize an RBP's sequence affinity in the form of PWMs (Section 3.3) usually rely on strong assumptions, such as the existence of a primary binding motif [3, 104, 153, 190]. Indeed, it is unclear whether the binding preference of many RBPs can be adequately represented by PWMs, as RBPs may harbor multiple binding domains or show affinity towards RNA secondary structure features. Further, RBPs may bind cooperatively or antagonistically, or as homo-or heterodimers, representing a higher-level binding syntax. Revealing the logical relationships between regulatory sequence elements may necessitate direct and repeated model probing [191], rather than the conventional post hoc analysis of feature attribution maps. Therefore, to fully understand the diverse modes of RBP-binding, novel global explainability approaches need to be devised, which take into account not only the nucleotide footprint of individual binding domains but also their interplay.

Current methods for protein-RNA interaction prediction aim to generalize to novel RNAs. This requires training data of the protein of interest, which must be obtained through experimental assays. While some studies showed that *in vitro* RBP-binding data can be used to train machine learning models that generalize to transcripts *in vivo*, methods that generalize to novel RBPs are desirable. However, the development of such method is severely hindered by the low number of profiled RNA-binding proteins. Among the over 2,000 proteins predicted to be RNA-binding [45], less than 200 are profiled with eCLIP experiments in the ENCODE database [71] as of September 2023. This scarcity makes training machine learning models challenging. To address this issue, one approach may involve extracting rich representations for a protein's amino acid sequence through structural modeling with AlphaFold [192] or large protein language models [193, 194], the latter of which has been shown to enable efficient fine-tuning on supervised tasks through few-shot learning [195].

Finally, recent advancements in the development of multiplexed CLIP assays have facilitated transcriptome-wide profiling of protein-RNA interactions involving multiple RBPs within a single experiment [84, 85]. This has the potential to significantly reduce costs while enabling large-scale profiling of protein-RNA interactions across various tissues and species. Such data will allow researchers to investigate how the tissue-specific presence of binding antagonists, or the absence of co-factors, affects the set of targets of a given RBP. Additionally, the recently introduced STAMP assay offers the capability to identify protein-RNA interactions at the single-cell level, providing valuable insights into the diversity of RNA processing across cell populations [196]. As larger datasets on the basis of these novel assays are gradually emerging, computational methods need to be adapted to accommodate them. Undoubtedly, exciting prospects await in the field of post-transcriptional RNA processing research, both for biologists and computational modelers alike.

BIBLIOGRAPHY

[1] Marc Horlacher, Svitlana Oleshko, Yue Hu, Mahsa Ghanbari, Giulia Cantini, Patrick Schinke, Ernesto Elorduy Vergara, Florian Bittner, Nikola S Mueller, Uwe Ohler, et al. "A computational map of the human-SARS-CoV-2 protein–RNA interactome predicted at single-nucleotide resolution." In: vol. 5. 1. Oxford University Press, 2023, lqad010.

[2] Marc Horlacher, Giulia Cantini, Julian Hesse, Patrick Schinke, Nicolas Goedert, Shubhankar Londhe, Lambert Moyon, and Annalisa Marsico. "A systematic benchmark of machine learning methods for protein–RNA interaction prediction." In: vol. 24. 5. Oxford University Press, 2023, bbad307.

[3] Marc Horlacher, Nils Wagner, Lambert Moyon, Klara Kuret, Nicolas Goedert, Marco Salvatore, Jernej Ule, Julien Gagneur, Ole Winther, and Annalisa Marsico. "Towards in silico CLIP-seq: predicting protein-RNA interaction via sequence-to-signal learning." In: vol. 24. 1. Springer, 2023, p. 180.

[4] Marco Salvatore, Marc Horlacher, Annalisa Marsico, Ole Winther, and Robin Andersson. "Transfer learning identifies sequence determinants of cell-type specific regulatory element accessibility." In: *NAR genomics and bioinformatics* 5.2 (2023), lqad026.

[5] Jun Wang, Marc Horlacher, Lixin Cheng, and Ole Winther. "RNA trafficking and subcellular localization-a review of mechanisms, experimental and predictive methodologies." In: *Briefings in bioinformatics* (2023), bbad249.

[6] Nora Schmidt, Sabina Ganskih, Yuanjie Wei, Alexander Gabel, Sebastian Zielinski, Hasmik Keshishian, Caleb A Lareau, Liv Zimmermann, Jana Makroczyova, Cadence Pearce, et al. "SND1 binds SARS-CoV-2 negative-sense RNA and promotes viral RNA synthesis through NSP9." In: *Cell* (2023).

[7] Nora Schmidt, Caleb A Lareau, Hasmik Keshishian, Sabina Ganskih, Cornelius Schneider, Thomas Hennig, Randy Melanson, Simone Werner, Yuanjie Wei, Matthias Zimmer, et al. "The SARS-CoV-2 RNA–protein interactome in infected human cells." In: *Nature microbiology* 6.3 (2021), pp. 339–353.

[8] Zhenghe Li and Peter D Nagy. "Diverse roles of host RNA binding proteins in RNA virus replication." In: *RNA biology* 8.2 (2011), pp. 305–315.

[9] Ryan A Flynn, Julia A Belk, Yanyan Qi, Yuki Yasumoto, Jin Wei, Mia Madel Alfajaro, Quanming Shi, Maxwell R Mumbach, Aditi Limaye, Peter C DeWeirdt, et al. "Discovery and functional interrogation of SARS-CoV-2 RNA-host protein interactions." In: *Cell* 184.9 (2021), pp. 2394–2411.

[10] Pablo Meyer and Julio Saez-Rodriguez. "Advances in systems biology modeling: 10 years of crowdsourcing DREAM challenges." In: *Cell Systems* 12.6 (2021), pp. 636–653.

[11] Jens Keilwagen, Stefan Posch, and Jan Grau. "Accurate prediction of cell type-specific transcription factor binding." In: *Genome biology* 20 (2019), pp. 1–17.

[12] Markus Hafner, Maria Katsantoni, Tino Köster, James Marks, Joyita Mukherjee, Dorothee Staiger, Jernej Ule, and Mihaela Zavolan. "CLIP and complementary methods." In: *Nature Reviews Methods Primers* 1.1 (2021), p. 20.

[13] Julian König, Kathi Zarnack, Gregor Rot, Tomaž Curk, Melis Kayikci, Blaž Zupan, Daniel J Turner, Nicholas M Luscombe, and Jernej Ule. "iCLIP reveals the function of hnRNP particles in splicing at individual nucleotide resolution." In: *Nature structural & molecular biology* 17.7 (2010), pp. 909–915.

[14] Eric L Van Nostrand, Gabriel A Pratt, Alexander A Shishkin, Chelsea Gelboin-Burkhart, Mark Y Fang, Balaji Sundararaman, Steven M Blue, Thai B Nguyen, Christine Surka, Keri Elkins, et al. "Robust transcriptome-wide discovery of RNA-binding protein binding sites with enhanced CLIP (eCLIP)." In: *Nature methods* 13.6 (2016), pp. 508–514.

[15] Marco Antonio Valencia-Sanchez, Jidong Liu, Gregory J Hannon, and Roy Parker. "Control of translation and mRNA degradation by miRNAs and siRNAs." In: *Genes & development* 20.5 (2006), pp. 515–524.

[16] Lasse S Kristensen, Maria S Andersen, Lotte VW Stagsted, Karoline K Ebbesen, Thomas B Hansen, and Jørgen Kjems. "The biogenesis, biology and characterization of circular RNAs." In: *Nature Reviews Genetics* 20.11 (2019), pp. 675–691.

[17] Wenlun Wang, Lu Min, Xinyuan Qiu, Xiaomin Wu, Chuanyang Liu, Jiaxin Ma, Dongyi Zhang, and Lingyun Zhu. "Biological function of long non-coding RNA (LncRNA) Xist." In: *Frontiers in cell and developmental biology* 9 (2021), p. 645647.

[18] Fabrizio Ferre, Alessio Colantoni, and Manuela Helmer-Citterich. "Revealing protein–lncRNA interaction." In: *Briefings in bioinformatics* 17.1 (2016), pp. 106–116.

[19] Andrew D Garst, Andrea L Edwards, and Robert T Batey. "Riboswitches: structures and mechanisms." In: *Cold Spring Harbor perspectives in biology* 3.6 (2011), a003533.

[20] Paul G Higgs and Niles Lehman. "The RNA World: molecular cooperation at the origins of life." In: *Nature Reviews Genetics* 16.1 (2015), pp. 7–17.

[21] Taifeng Wu and Leslie E Orgel. "Nonenzymic template-directed synthesis on oligodeoxycytidylate sequences in hairpin oligonucleotides." In: *Journal of the American Chemical Society* 114.1 (1992), pp. 317–322.

[22] Sandy L Klemm, Zohar Shipony, and William J Greenleaf. "Chromatin accessibility and the regulatory epigenome." In: *Nature Reviews Genetics* 20.4 (2019), pp. 207–220.

[23] Tina Glisovic, Jennifer L Bachorik, Jeongsik Yong, and Gideon Dreyfuss. "RNA-binding proteins and post-transcriptional gene regulation." In: *FEBS letters* 582.14 (2008), pp. 1977–1986.

[24] Anand Ramanathan, G Brett Robb, and Siu-Hong Chan. "mRNA capping: biological functions and applications." In: *Nucleic acids research* 44.16 (2016), pp. 7511–7526.

[25] Shin Moteki and David Price. "Functional coupling of capping and transcription of mRNA." In: *Molecular cell* 10.3 (2002), pp. 599–609.

[26] Sibylle Mitschka and Christine Mayr. "Context-specific regulation and function of mRNA alternative polyadenylation." In: *Nature Reviews Molecular Cell Biology* 23.12 (2022), pp. 779–796.

[27] Andreas J Gruber and Mihaela Zavolan. "Alternative cleavage and polyadenylation in health and disease." In: *Nature Reviews Genetics* 20.10 (2019), pp. 599–614.

[28] Francisco E Baralle and Jimena Giudice. "Alternative splicing as a regulator of development and tissue identity." In: *Nature reviews Molecular cell biology* 18.7 (2017), pp. 437–451.

[29] Robert K Bradley and Olga Anczuków. "RNA splicing dysregulation and the hallmarks of cancer." In: *Nature Reviews Cancer* 23.3 (2023), pp. 135–155.

[30] Nils Wagner, Muhammed H Çelik, Florian R Hölzlwimmer, Christian Mertes, Holger Prokisch, Vicente A Yépez, and Julien Gagneur. "Aberrant splicing prediction across human tissues." In: *Nature Genetics* (2023), pp. 1–10.

[31] Christopher R Neil, Michael W Seiler, Dominic J Reynolds, Jesse J Smith, Frédéric H Vaillancourt, Peter G Smith, and Anant A Agrawal. "Reprogramming RNA processing: an emerging therapeutic landscape." In: *Trends in Pharmacological Sciences* (2022).

[32] Sulagna Das, Maria Vera, Valentina Gandin, Robert H Singer, and Evelina Tutucci. "Intracellular mRNA transport and localized translation." In: *Nature Reviews Molecular Cell Biology* 22.7 (2021), pp. 483–504.

[33] Michael S Fernandopulle, Jennifer Lippincott-Schwartz, and Michael E Ward. "RNA transport and local translation in neurodevelopmental and neurodegenerative disease." In: *Nature neuroscience* 24.5 (2021), pp. 622–632.

[34] Ya-Cheng Liao, Michael S Fernandopulle, Guozhen Wang, Heejun Choi, Ling Hao, Catherine M Drerup, Rajan Patel, Seema Qamar, Jonathon Nixon-Abell, Yi Shen, et al. "RNA granules hitchhike on lysosomes for long-distance transport, using annexin A11 as a molecular tether." In: *Cell* 179.1 (2019), pp. 147–164.

[35] Sebastian Baumann, Julian König, Janine Koepke, and Michael Feldbrügge. "Endosomal transport of septin mRNA and protein indicates local translation on endosomes and is required for correct septin filamentation." In: *EMBO reports* 15.1 (2014), pp. 94–102.

[36] Daniel R Schoenberg and Lynne E Maquat. "Regulation of cytoplasmic mRNA decay." In: *Nature Reviews Genetics* 13.4 (2012), pp. 246–259.

[37] Mingqiang Deng, Xiwei Wang, Zhi Xiong, and Peng Tang. "Control of RNA degradation in cell fate decision." In: *Frontiers in Cell and Developmental Biology* 11 (2023), p. 1164546.

[38] Jonathan Houseley and David Tollervey. "The many pathways of RNA degradation." In: *Cell* 136.4 (2009), pp. 763–776.

[39] Boxuan Simen Zhao, Ian A Roundtree, and Chuan He. "Post-transcriptional gene regulation by mRNA modifications." In: *Nature reviews Molecular cell biology* 18.1 (2017), pp. 31–42.

[40] Ying Yang, Phillip J Hsu, Yu-Sheng Chen, and Yun-Gui Yang. “Dynamic transcriptomic m6A decoration: writers, erasers, readers and functions in RNA metabolism.” In: *Cell research* 28.6 (2018), pp. 616–624.

[41] Wei Zhang, Yang Qian, and Guifang Jia. “The detection and functions of RNA modification m6A based on m6A writers and erasers.” In: *Journal of Biological Chemistry* 297.2 (2021).

[42] Yuanhui Mao, Leiming Dong, Xiao-Min Liu, Jiayin Guo, Honghui Ma, Bin Shen, and Shu-Bing Qian. “m6A in mRNA coding regions promotes translation via the RNA helicase-containing YTHDC2.” In: *Nature communications* 10.1 (2019), p. 5332.

[43] Xiulin Jiang, Baiyang Liu, Zhi Nie, Lincan Duan, Qiuxia Xiong, Zhixian Jin, Cuiping Yang, and Yongbin Chen. “The role of m6A modification in the biological functions and diseases.” In: *Signal transduction and targeted therapy* 6.1 (2021), p. 74.

[44] Stefanie Gerstberger, Markus Hafner, and Thomas Tuschl. “A census of human RNA-binding proteins.” In: *Nature Reviews Genetics* 15.12 (2014), pp. 829–845.

[45] Fátima Gebauer, Thomas Schwarzl, Juan Valcárcel, and Matthias W Hentze. “RNA-binding proteins in human genetic disease.” In: *Nature Reviews Genetics* 22.3 (2021), pp. 185–198.

[46] Matthias W Hentze, Alfredo Castello, Thomas Schwarzl, and Thomas Preiss. “A brave new world of RNA-binding proteins.” In: *Nature reviews Molecular cell biology* 19.5 (2018), pp. 327–341.

[47] Meredith Corley, Margaret C Burns, and Gene W Yeo. “How RNA-binding proteins interact with RNA: molecules and mechanisms.” In: *Molecular cell* 78.1 (2020), pp. 9–29.

[48] Wenhao Jin, Kristopher W Brannan, Katannya Kapeli, Samuel S Park, Hui Qing Tan, Maya L Gosztyla, Mayuresh Mujumdar, Joshua Ahdout, Bryce Henroid, Katherine Rothamel, et al. “HydRA: Deep-learning models for predicting RNA-binding capacity from protein interaction association context and protein sequence.” In: *bioRxiv* (2022), pp. 2022–12.

[49] Antoine Cléry, Markus Blatter, and Frédéric HT Allain. “RNA recognition motifs: boring? Not quite.” In: *Current opinion in structural biology* 18.3 (2008), pp. 290–298.

[50] Alfredo Castello, Bernd Fischer, Christian K Frese, Rastislav Horos, Anne-Marie Alleaume, Sophia Foehr, Tomaz Curk, Jeroen Krijgsveld, and Matthias W Hentze. “Comprehensive identification of RNA-binding domains in human cells.” In: *Molecular cell* 63.4 (2016), pp. 696–710.

[51] Waleed S Albihlal and André P Gerber. “Unconventional RNA-binding proteins: an uncharted zone in RNA biology.” In: *FEBS letters* 592.17 (2018), pp. 2917–2931.

[52] Xiang-Dong Fu and Manuel Ares Jr. “Context-dependent control of alternative splicing by RNA-binding proteins.” In: *Nature Reviews Genetics* 15.10 (2014), pp. 689–701.

[53] Surender Vashist, Luis Urena, Yasmin Chaudhry, and Ian Goodfellow. "Identification of RNA-protein interaction networks involved in the norovirus life cycle." In: *Journal of virology* 86.22 (2012), pp. 11977–11990.

[54] Manuel Garcia-Moreno, Aino I Järvelin, and Alfredo Castello. "Unconventional RNA-binding proteins step into the virus–host battlefront." In: *Wiley Interdisciplinary Reviews: RNA* 9.6 (2018), e1498.

[55] Jerome M Molleston and Sara Cherry. "Attacked from all sides: RNA decay in antiviral defense." In: *Viruses* 9.1 (2017), p. 2.

[56] Donny D Licatalosi, Xuan Ye, and Eckhard Jankowsky. "Approaches for measuring the dynamics of RNA–protein interactions." In: *Wiley Interdisciplinary Reviews: RNA* 11.1 (2020), e1565.

[57] Colleen A McHugh, Pamela Russell, and Mitchell Guttman. "Methods for comprehensive experimental identification of RNA-protein interactions." In: *Genome biology* 15 (2014), pp. 1–10.

[58] Colleen A McHugh, Chun-Kan Chen, Amy Chow, Christine F Surka, Christina Tran, Patrick McDonel, Amy Pandya-Jones, Mario Blanco, Christina Burghard, Annie Moradian, et al. "The Xist lncRNA interacts directly with SHARP to silence transcription through HDAC3." In: *Nature* 521.7551 (2015), pp. 232–236.

[59] Alfredo Castello, Bernd Fischer, Katrin Eichelbaum, Rastislav Horos, Benedikt M Beckmann, Claudia Strein, Norman E Davey, David T Humphreys, Thomas Preiss, Lars M Steinmetz, et al. "Insights into RNA biology from an atlas of mammalian mRNA-binding proteins." In: *Cell* 149.6 (2012), pp. 1393–1406.

[60] Alexander G Baltz, Mathias Munschauer, Björn Schwanhäusser, Alexandra Vasile, Yasuhiro Murakawa, Markus Schueler, Noah Youngs, Duncan Penfold-Brown, Kevin Drew, Miha Milek, et al. "The mRNA-bound proteome and its global occupancy profile on protein-coding transcripts." In: *Molecular cell* 46.5 (2012), pp. 674–690.

[61] Joel I Perez-Perri, Birgit Rogell, Thomas Schwarzl, Frank Stein, Yang Zhou, Mandy Rettel, Annika Brosig, and Matthias W Hentze. "Discovery of RNA-binding proteins and characterization of their dynamic responses by enhanced RNA interactome capture." In: *Nature communications* 9.1 (2018), p. 4408.

[62] *RBPbase - a comprehensive database of eukaryotic RNA-binding proteins (RBPs) with their RBP annotations*. URL: https://apps.embl.de/rbpbase/ (visited on 09/01/2023).

[63] Ci Chu, Qiangfeng Cliff Zhang, Simão Teixeira Da Rocha, Ryan A Flynn, Maheetha Bharadwaj, J Mauro Calabrese, Terry Magnuson, Edith Heard, and Howard Y Chang. "Systematic discovery of Xist RNA binding proteins." In: *Cell* 161.2 (2015), pp. 404–416.

[64] Athéna Labeau, Luc Fery-Simonian, Alain Lefevre-Utile, Marie Pourcelot, Lucie Bonnet-Madin, Vassili Soumelis, Vincent Lotteau, Pierre-Olivier Vidalain, Ali Amara, and Laurent Meertens. "Characterization and functional interrogation of the SARS-CoV-2 RNA interactome." In: *Cell reports* 39.4 (2022).

[65] Debashish Ray, Hilal Kazan, Esther T Chan, Lourdes Pena Castillo, Sidharth Chaudhry, Shaheynoor Talukder, Benjamin J Blencowe, Quaid Morris, and Timothy R Hughes. "Rapid and systematic analysis of the RNA recognition specificities of RNA-binding proteins." In: *Nature biotechnology* 27.7 (2009), pp. 667–670.

[66] Nicole J Lambert, Alex D Robertson, and Christopher B Burge. "RNA Bind-n-Seq: measuring the binding affinity landscape of RNA-binding proteins." In: *Methods in enzymology*. Vol. 558. Elsevier, 2015, pp. 465–493.

[67] Arttu Jolma, Jilin Zhang, Estefania Mondragón, Ekaterina Morgunova, Teemu Kivioja, Kaitlin U Laverty, Yimeng Yin, Fangjie Zhu, Gleb Bourenkov, Quaid Morris, et al. "Binding specificities of human RNA-binding proteins toward structured and linear RNA sequences." In: *Genome research* 30.7 (2020), pp. 962–973.

[68] Jernej Ule, Kirk B Jensen, Matteo Ruggiu, Aldo Mele, Aljaz Ule, and Robert B Darnell. "CLIP identifies Nova-regulated RNA networks in the brain." In: *Science* 302.5648 (2003), pp. 1212–1215.

[69] Charles Danan, Sudhir Manickavel, and Markus Hafner. "PAR-CLIP: a method for transcriptome-wide identification of RNA binding protein interaction sites." In: *Post-Transcriptional Gene Regulation* (2016), pp. 153–173.

[70] Ina Huppertz, Jan Attig, Andrea D'Ambrogio, Laura E Easton, Christopher R Sibley, Yoichiro Sugimoto, Mojca Tajnik, Julian König, and Jernej Ule. "iCLIP: protein–RNA interactions at nucleotide resolution." In: *Methods* 65.3 (2014), pp. 274–287.

[71] ENCODE Project Consortium et al. "An integrated encyclopedia of DNA elements in the human genome." In: *Nature* 489.7414 (2012), p. 57.

[72] Emily C Wheeler, Eric L Van Nostrand, and Gene W Yeo. "Advances and challenges in the detection of transcriptome-wide protein–RNA interactions." In: *Wiley Interdisciplinary Reviews: RNA* 9.1 (2018), e1436.

[73] Klara Kuret, Aram Gustav Amalietti, D Marc Jones, Charlotte Capitanchik, and Jernej Ule. "Positional motif analysis reveals the extent of specificity of protein-RNA interactions observed by CLIP." In: *Genome biology* 23.1 (2022), p. 191.

[74] Eric L Van Nostrand, Peter Freese, Gabriel A Pratt, Xiaofeng Wang, Xintao Wei, Rui Xiao, Steven M Blue, Jia-Yu Chen, Neal AL Cody, Daniel Dominguez, et al. "A large-scale binding and functional map of human RNA-binding proteins." In: *Nature* 583.7818 (2020), pp. 711–719.

[75] Nejc Haberman, Ina Huppertz, Jan Attig, Julian König, Zhen Wang, Christian Hauer, Matthias W Hentze, Andreas E Kulozik, Hervé Le Hir, Tomaž Curk, et al. "Insights into the design and interpretation of iCLIP experiments." In: *Genome biology* 18 (2017), pp. 1–21.

[76] Yoichiro Sugimoto, Julian König, Shobbir Hussain, Blaž Zupan, Tomaž Curk, Michaela Frye, and Jernej Ule. "Analysis of CLIP and iCLIP methods for nucleotide-resolution studies of protein-RNA interactions." In: *Genome biology* 13.8 (2012), pp. 1–13.

[77] Michael T Lovci, Dana Ghanem, Henry Marr, Justin Arnold, Sherry Gee, Marilyn Parra, Tiffany Y Liang, Thomas J Stark, Lauren T Gehman, Shawn Hoon, et al. "Rbfox proteins regulate alternative mRNA splicing through evolutionarily conserved RNA bridges." In: *Nature structural & molecular biology* 20.12 (2013), pp. 1434–1442.

[78] YeoLab. *CLIPper (Github)*. 2020. URL: https://github.com/YeoLab/clipper (visited on 09/30/2023).

[79] David L Corcoran, Stoyan Georgiev, Neelanjan Mukherjee, Eva Gottwein, Rebecca L Skalsky, Jack D Keene, and Uwe Ohler. "PARalyzer: definition of RNA binding sites from PAR-CLIP short-read sequence data." In: *Genome biology* 12 (2011), pp. 1–16.

[80] Philip J Uren, Emad Bahrami-Samani, Suzanne C Burns, Mei Qiao, Fedor V Karginov, Emily Hodges, Gregory J Hannon, Jeremy R Sanford, Luiz OF Penalva, and Andrew D Smith. "Site identification in high-throughput RNA–protein interaction data." In: *Bioinformatics* 28.23 (2012), pp. 3013–3020.

[81] UleLab. *clippy (Github)*. 2023. URL: https://github.com/ulelab/clippy (visited on 09/30/2023).

[82] Sabrina Krakau. "Statistical models to capture protein-RNA interaction footprints from truncation-based CLIP-seq data." PhD thesis. 2020.

[83] Sabrina Krakau, Hugues Richard, and Annalisa Marsico. "PureCLIP: capturing target-specific protein–RNA interaction footprints from single-nucleotide CLIP-seq data." In: *Genome biology* 18 (2017), pp. 1–17.

[84] Erica Wolin, Jimmy K Guo, Mario R Blanco, Andrew A Perez, Isabel N Goronzy, Ahmed A Abdou, Darvesh Gorhe, Mitchell Guttman, and Marko Jovanovic. "SPIDR: a highly multiplexed method for mapping RNA-protein interactions uncovers a potential mechanism for selective translational suppression upon cellular stress." In: *bioRxiv* (2023), pp. 2023–06.

[85] Daniel A Lorenz, Hsuan-Lin Her, Kylie A Shen, Katie Rothamel, Kasey R Hutt, Allan C Nojadera, Stephanie C Bruns, Sergei A Manakov, Brian A Yee, Karen B Chapman, et al. "Multiplexed transcriptome discovery of RNA-binding protein binding sites by antibody-barcode eCLIP." In: *Nature Methods* 20.1 (2023), pp. 65–69.

[86] Gerd Anders, Sebastian D Mackowiak, Marvin Jens, Jonas Maaskola, Andreas Kuntzagk, Nikolaus Rajewsky, Markus Landthaler, and Christoph Dieterich. "doRiNA: a database of RNA interactions in post-transcriptional regulation." In: *Nucleic acids research* 40.D1 (2012), pp. D180–D186.

[87] Kai Blin, Christoph Dieterich, Ricardo Wurmus, Nikolaus Rajewsky, Markus Landthaler, and Altuna Akalin. "DoRiNA 2.0—upgrading the doRiNA database of RNA interactions in post-transcriptional regulation." In: *Nucleic acids research* 43.D1 (2015), pp. D160–D167.

[88] Martin Stražar, Marinka Žitnik, Blaž Zupan, Jernej Ule, and Tomaž Curk. "Orthogonal matrix factorization enables integrative analysis of multiple RNA binding proteins." In: *Bioinformatics* 32.10 (2016), pp. 1527–1535.

[89] Inbal Paz, Idit Kosti, Manuel Ares Jr, Melissa Cline, and Yael Mandel-Gutfreund. "RBPmap: a web server for mapping binding sites of RNA-binding proteins." In: *Nucleic acids research* 42.W1 (2014), W361–W367.

[90] Kate B Cook, Hilal Kazan, Khalid Zuberi, Quaid Morris, and Timothy R Hughes. "RBPDB: a database of RNA-binding specificities." In: *Nucleic acids research* 39.suppl_1 (2010), pp. D301–D308.

[91] Girolamo Giudice, Fátima Sánchez-Cabo, Carlos Torroja, and Enrique Lara-Pezzi. "ATtRACT—a database of RNA-binding proteins and associated motifs." In: *Database* 2016 (2016), baw035.

[92] Debashish Ray, Hilal Kazan, Kate B Cook, Matthew T Weirauch, Hamed S Najafabadi, Xiao Li, Serge Gueroussov, Mihai Albu, Hong Zheng, Ally Yang, et al. "A compendium of RNA-binding motifs for decoding gene regulation." In: *Nature* 499.7457 (2013), pp. 172–177.

[93] Huijuan Feng, Suying Bao, Mohammad Alinoor Rahman, Sebastien M Weyn-Vanhentenryck, Aziz Khan, Justin Wong, Ankeeta Shah, Elise D Flynn, Adrian R Krainer, and Chaolin Zhang. "Modeling RNA-binding protein specificity in vivo by precisely registering protein-RNA crosslink sites." In: *Molecular cell* 74.6 (2019), pp. 1189–1204.

[94] Louis Philip Benoit Bouvrette, Samantha Bovaird, Mathieu Blanchette, and Eric Lécuyer. "oRNAment: a database of putative RNA binding protein target sites in the transcriptomes of model species." In: *Nucleic acids research* 48.D1 (2020), pp. D166–D173.

[95] Timothy L Bailey, James Johnson, Charles E Grant, and William S Noble. "The MEME suite." In: *Nucleic acids research* 43.W1 (2015), W39–W49.

[96] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep learning*. MIT press, 2016.

[97] Francois Chollet. *Deep learning with Python*. Simon and Schuster, 2021.

[98] Frank Rosenblatt. "The perceptron: a probabilistic model for information storage and organization in the brain." In: *Psychological review* 65.6 (1958), p. 386.

[99] Raúl Rojas. *Neural networks: a systematic introduction*. Springer Science & Business Media, 2013.

[100] Kurt Hornik, Maxwell Stinchcombe, and Halbert White. "Multilayer feedforward networks are universal approximators." In: *Neural networks* 2.5 (1989), pp. 359–366.

[101] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. "Gradient-based learning applied to document recognition." In: *Proceedings of the IEEE* 86.11 (1998), pp. 2278–2324.

[102] Qi Wang, Yue Ma, Kun Zhao, and Yingjie Tian. "A comprehensive survey of loss functions in machine learning." In: *Annals of Data Science* (2020), pp. 1–26.

[103] Sebastian Ruder. "An overview of gradient descent optimization algorithms." In: *arXiv preprint arXiv:1609.04747* (2016).

[104] Stefan Budach and Annalisa Marsico. “Pysster: classification of biological sequences by learning sequence and structure motifs with convolutional neural networks.” In: *Bioinformatics* 34.17 (2018), pp. 3035–3037.

[105] Diederik P Kingma and Jimmy Ba. “Adam: A method for stochastic optimization.” In: *arXiv preprint arXiv:1412.6980* (2014).

[106] Yoshua Bengio, Patrice Simard, and Paolo Frasconi. “Learning long-term dependencies with gradient descent is difficult.” In: *IEEE transactions on neural networks* 5.2 (1994), pp. 157–166.

[107] Lu Lu, Yeonjong Shin, Yanhui Su, and George Em Karniadakis. “Dying relu and initialization: Theory and numerical examples.” In: *arXiv preprint arXiv:1903.06733* (2019).

[108] Dan Hendrycks and Kevin Gimpel. “Gaussian error linear units (gelus).” In: *arXiv preprint arXiv:1606.08415* (2016).

[109] Prajit Ramachandran, Barret Zoph, and Quoc V Le. “Searching for activation functions.” In: *arXiv preprint arXiv:1710.05941* (2017).

[110] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. “Imagenet: A large-scale hierarchical image database.” In: *2009 IEEE conference on computer vision and pattern recognition*. Ieee. 2009, pp. 248–255.

[111] Diganta Misra. “Mish: A self regularized non-monotonic activation function.” In: *arXiv preprint arXiv:1908.08681* (2019).

[112] Žiga Avsec, Melanie Weilert, Avanti Shrikumar, Sabrina Krueger, Amr Alexandari, Khyati Dalal, Robin Fropf, Charles McAnany, Julien Gagneur, Anshul Kundaje, et al. “Base-resolution models of transcription-factor binding reveal soft motif syntax.” In: *Nature Genetics* 53.3 (2021), pp. 354–366.

[113] Kaelan J Brennan, Melanie Weilert, Sabrina Krueger, Anusri Pampari, Hsiao-Yun Liu, Ally WH Yang, Timothy R Hughes, Christine A Rushlow, Anshul Kundaje, and Julia Zeitlinger. “Chromatin accessibility is a two-tier process regulated by transcription factor pioneering and enhancer activation.” In: *bioRxiv* (2022), pp. 2022–12.

[114] Babak Alipanahi, Andrew Delong, Matthew T Weirauch, and Brendan J Frey. “Predicting the sequence specificities of DNA-and RNA-binding proteins by deep learning.” In: *Nature biotechnology* 33.8 (2015), pp. 831–838.

[115] David R Kelley, Jasper Snoek, and John L Rinn. “Basset: learning the regulatory code of the accessible genome with deep convolutional neural networks.” In: *Genome research* 26.7 (2016), pp. 990–999.

[116] Fisher Yu and Vladlen Koltun. “Multi-scale context aggregation by dilated convolutions.” In: *arXiv preprint arXiv:1511.07122* (2015).

[117] Dong-Hwan Jang, Sanghyeok Chu, Joonhyuk Kim, and Bohyung Han. “Pooling revisited: Your receptive field is suboptimal.” In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022, pp. 549–558.

[118] G Benegas, SS Batra, and YS Song. “DNA language models are powerful zero-shot predictors of non-coding variant effects.” In: (2022).

[119] Trevor Hastie, Robert Tibshirani, Jerome H Friedman, and Jerome H Friedman. *The elements of statistical learning: data mining, inference, and prediction*. Vol. 2. Springer, 2009.

[120] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. "Dropout: a simple way to prevent neural networks from overfitting." In: *The journal of machine learning research* 15.1 (2014), pp. 1929–1958.

[121] Sergey Ioffe and Christian Szegedy. "Batch normalization: Accelerating deep network training by reducing internal covariate shift." In: *International conference on machine learning*. pmlr. 2015, pp. 448–456.

[122] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. "Deep residual learning for image recognition." In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 770–778.

[123] Xavier Glorot and Yoshua Bengio. "Understanding the difficulty of training deep feedforward neural networks." In: *Proceedings of the thirteenth international conference on artificial intelligence and statistics*. JMLR Workshop and Conference Proceedings. 2010, pp. 249–256.

[124] Alejandro Barredo Arrieta, Natalia Díaz-Rodríguez, Javier Del Ser, Adrien Bennetot, Siham Tabik, Alberto Barbado, Salvador García, Sergio Gil-López, Daniel Molina, Richard Benjamins, et al. "Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI." In: *Information fusion* 58 (2020), pp. 82–115.

[125] Leo Breiman. "Random forests." In: *Machine learning* 45 (2001), pp. 5–32.

[126] Lipo Wang. *Support vector machines: theory and applications*. Vol. 177. Springer Science & Business Media, 2005.

[127] Rabia Saleem, Bo Yuan, Fatih Kurugollu, Ashiq Anjum, and Lu Liu. "Explaining deep neural networks: A survey on the global interpretation methods." In: *Neurocomputing* (2022).

[128] Divyanshi Srivastava, Begüm Aydin, Esteban O Mazzoni, and Shaun Mahony. "An interpretable bimodal neural network characterizes the sequence and preexisting chromatin predictors of induced transcription factor binding." In: *Genome biology* 22.1 (2021), pp. 1–25.

[129] Matthew D Zeiler and Rob Fergus. "Visualizing and understanding convolutional networks." In: *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part I 13*. Springer. 2014, pp. 818–833.

[130] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. "Deep inside convolutional networks: Visualising image classification models and saliency maps." In: *arXiv preprint arXiv:1312.6034* (2013).

[131] Marco Ancona, Enea Ceolini, Cengiz Öztireli, and Markus Gross. "Towards better understanding of gradient-based attribution methods for deep neural networks." In: *arXiv preprint arXiv:1711.06104* (2017).

[132] Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. "Learning important features through propagating activation differences." In: *International conference on machine learning*. PMLR. 2017, pp. 3145–3153.

[133] Xiaoyong Pan, Peter Rijnbeek, Junchi Yan, and Hong-Bin Shen. "Prediction of RNA-protein sequence and structure binding preferences using deep convolutional and recurrent neural networks." In: *BMC genomics* 19.1 (2018), pp. 1–11.

[134] Ameni Trabelsi, Mohamed Chaabane, and Asa Ben-Hur. "Comprehensive evaluation of deep learning architectures for prediction of DNA/RNA sequence binding specificities." In: *Bioinformatics* 35.14 (2019), pp. i269–i277.

[135] Alexander Gulliver Bjørnholt Grønning, Thomas Koed Doktor, Simon Jonas Larsen, Ulrika Simone Spangsberg Petersen, Lise Lolle Holm, Gitte Hoffmann Bruun, Michael Birkerod Hansen, Anne-Mette Hartung, Jan Baumbach, and Brage Storstein Andresen. "DeepCLIP: predicting the effect of mutations on protein–RNA binding with deep learning." In: *Nucleic acids research* 48.13 (2020), pp. 7099–7118.

[136] Stefan Budach. "Explainable deep learning models for biological sequence classification." PhD thesis. 2021.

[137] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. "Bert: Pre-training of deep bidirectional transformers for language understanding." In: *arXiv preprint arXiv:1810.04805* (2018).

[138] Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, et al. "Opt: Open pre-trained transformer language models." In: *arXiv preprint arXiv:2205.01068* (2022).

[139] Rohan Anil, Andrew M Dai, Orhan Firat, Melvin Johnson, Dmitry Lepikhin, Alexandre Passos, Siamak Shakeri, Emanuel Taropa, Paige Bailey, Zhifeng Chen, et al. "Palm 2 technical report." In: *arXiv preprint arXiv:2305.10403* (2023).

[140] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. "Llama: Open and efficient foundation language models." In: *arXiv preprint arXiv:2302.13971* (2023).

[141] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. "An image is worth 16x16 words: Transformers for image recognition at scale." In: *arXiv preprint arXiv:2010.11929* (2020).

[142] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. "Attention is all you need." In: *Advances in neural information processing systems* 30 (2017).

[143] Yanrong Ji, Zhihan Zhou, Han Liu, and Ramana V Davuluri. "DNABERT: pre-trained Bidirectional Encoder Representations from Transformers model for DNA-language in genome." In: *Bioinformatics* 37.15 (2021), pp. 2112–2120.

[144] Keisuke Yamada and Michiaki Hamada. "Prediction of RNA–protein interactions using a nucleotide language model." In: *Bioinformatics Advances* 2.1 (2022), vbac023.

[145] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. "Axiomatic attribution for deep networks." In: *International conference on machine learning*. PMLR. 2017, pp. 3319–3328.

[146] Christoph Molnar. *Interpretable Machine Learning. A Guide for Making Black Box Models Explainable*. 2nd ed. 2022. URL: https://christophm.github.io/interpretable-ml-book.

[147] Daniel Smilkov, Nikhil Thorat, Been Kim, Fernanda Viégas, and Martin Wattenberg. "Smoothgrad: removing noise by adding noise." In: *arXiv preprint arXiv:1706.03825* (2017).

[148] Avanti Shrikumar, Peyton Greenside, Anna Shcherbina, and Anshul Kundaje. "Not just a black box: Learning important features through propagating activation differences." In: *arXiv preprint arXiv:1605.01713* (2016).

[149] Žiga Avsec, Vikram Agarwal, Daniel Visentin, Joseph R Ledsam, Agnieszka Grabska-Barwinska, Kyle R Taylor, Yannis Assael, John Jumper, Pushmeet Kohli, and David R Kelley. "Effective gene expression prediction from sequence by integrating long-range interactions." In: *Nature methods* 18.10 (2021), pp. 1196–1203.

[150] Michael Uhl, Van Dinh Tran, Florian Heyl, and Rolf Backofen. "RNAProt: an efficient and feature-rich RNA binding protein binding site predictor." In: *GigaScience* 10.8 (2021), giab054.

[151] Lei Sun, Kui Xu, Wenze Huang, Yucheng T Yang, Pan Li, Lei Tang, Tuanlin Xiong, and Qiangfeng Cliff Zhang. "Predicting dynamic cellular protein–RNA interactions by deep learning using in vivo RNA structures." In: *Cell research* 31.5 (2021), pp. 495–516.

[152] Ryan A Flynn, Qiangfeng Cliff Zhang, Robert C Spitale, Byron Lee, Maxwell R Mumbach, and Howard Y Chang. "Transcriptome-wide interrogation of RNA secondary structure in living cells with icSHAPE." In: *Nature protocols* 11.2 (2016), pp. 273–290.

[153] Mahsa Ghanbari and Uwe Ohler. "Deep neural networks for interpreting RNA-binding protein target preferences." In: *Genome research* 30.2 (2020), pp. 214–226.

[154] Shitao Zhao and Michiaki Hamada. "Multi-resBind: a residual network-based multi-label classifier for in vivo RNA binding prediction and preference visualization." In: *BMC bioinformatics* 22.1 (2021), pp. 1–15.

[155] Jonathan Karin, Hagai Michel, and Yaron Orenstein. "Multirbp: multi-task neural network for protein-RNA binding prediction." In: *Proceedings of the 12th ACM conference on bioinformatics, computational biology, and health informatics*. 2021, pp. 1–9.

[156] Ryan R Wick, Louise M Judd, and Kathryn E Holt. "Performance of neural network basecalling tools for Oxford Nanopore sequencing." In: *Genome biology* 20 (2019), pp. 1–10.

[157] Ryan Poplin, Pi-Chuan Chang, David Alexander, Scott Schwartz, Thomas Colthurst, Alexander Ku, Dan Newburger, Jojo Dijamco, Nam Nguyen, Pegah T Afshar, et al. "A universal SNP and small-indel variant caller using deep neural networks." In: *Nature biotechnology* 36.10 (2018), pp. 983–987.

[158] Peter K Koo and Matt Ploenzke. "Deep learning for inferring transcription factor binding sites." In: *Current opinion in systems biology* 19 (2020), pp. 16–23.

[159] Qijin Yin, Mengmeng Wu, Qiao Liu, Hairong Lv, and Rui Jiang. "DeepHistone: a deep learning approach to predicting histone modifications." In: *BMC genomics* 20.2 (2019), pp. 11–23.

[160] Christof Angermueller, Heather J Lee, Wolf Reik, and Oliver Stegle. "DeepCpG: accurate prediction of single-cell DNA methylation states using deep learning." In: *Genome biology* 18.1 (2017), pp. 1–13.

[161] Kengo Sato, Manato Akiyama, and Yasubumi Sakakibara. "RNA secondary structure prediction using deep learning with thermodynamic integration." In: *Nature communications* 12.1 (2021), p. 941.

[162] Tao Shen, Zhihang Hu, Zhangzhi Peng, Jiayang Chen, Peng Xiong, Liang Hong, Liangzhen Zheng, Yixuan Wang, Irwin King, Sheng Wang, et al. "E2Efold-3D: end-to-end deep learning method for accurate de novo RNA 3D structure prediction." In: *arXiv preprint arXiv:2207.01586* (2022).

[163] Kishore Jaganathan, Sofia Kyriazopoulou Panagiotopoulou, Jeremy F McRae, Siavash Fazel Darbandi, David Knowles, Yang I Li, Jack A Kosmicki, Juan Arbelaez, Wenwu Cui, Grace B Schwartz, et al. "Predicting splicing from primary sequence with deep learning." In: *Cell* 176.3 (2019), pp. 535–548.

[164] Vikram Agarwal and David R Kelley. "The genetic and biochemical determinants of mRNA degradation rates in mammals." In: *Genome Biology* 23.1 (2022), p. 245.

[165] Duolin Wang, Zhaoyue Zhang, Yuexu Jiang, Ziting Mao, Dong Wang, Hao Lin, and Dong Xu. "DM3Loc: multi-label mRNA subcellular localization prediction and analysis based on multi-head self-attention mechanism." In: *Nucleic acids research* 49.8 (2021), e46–e46.

[166] Alexander Karollus, Žiga Avsec, and Julien Gagneur. "Predicting mean ribosome load for 5′UTR of any length using deep learning." In: *PLoS computational biology* 17.5 (2021), e1008982.

[167] David R Kelley. "Cross-species regulatory sequence activity prediction." In: *PLoS computational biology* 16.7 (2020), e1008050.

[168] Hugo Dalla-Torre, Liam Gonzalez, Javier Mendoza-Revilla, Nicolas Lopez Carranza, Adam Henryk Grzywaczewski, Francesco Oteri, Christian Dallago, Evan Trop, Hassan Sirelkhatim, Guillaume Richard, et al. "The Nucleotide Transformer: Building and Evaluating Robust Foundation Models for Human Genomics." In: *bioRxiv* (2023), pp. 2023–01.

[169] Eric Nguyen, Michael Poli, Marjan Faizi, Armin Thomas, Callum Birch-Sykes, Michael Wornow, Aman Patel, Clayton Rabideau, Stefano Massaroli, Yoshua Bengio, et al. "Hyenadna: Long-range genomic sequence modeling at single nucleotide resolution." In: *arXiv preprint arXiv:2306.15794* (2023).

[170] Dennis Gankin, Alexander Karollus, Martin Grosshauser, Kristian Klemon, Johannes Hingerl, and Julien Gagneur. "Species-aware DNA language modeling." In: *bioRxiv* (2023), pp. 2023–01.

[171] Veniamin Fishman, Yuri Kuratov, Maxim Petrov, Aleksei Shmelev, Denis Shepelin, Nikolay Chekanov, Olga Kardymon, and Mikhail Burtsev. "GENA-LM: A Family of Open-Source Foundational Models for Long DNA Sequences." In: *bioRxiv* (2023), pp. 2023–06.

[172] Zeming Lin, Halil Akin, Roshan Rao, Brian Hie, Zhongkai Zhu, Wenting Lu, Nikita Smetanin, Robert Verkuil, Ori Kabeli, Yaniv Shmueli, et al. "Evolutionary-scale prediction of atomic-level protein structure with a language model." In: *Science* 379.6637 (2023), pp. 1123–1130.

[173] Ahmed Elnaggar et al. "ProtTrans: Toward Understanding the Language of Life Through Self-Supervised Learning." In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 44.10 (2022), pp. 7112–7127. DOI: 10.1109/TPAMI.2021.3095381.

[174] Jian Zhou and Olga G Troyanskaya. "Predicting effects of noncoding variants with deep learning–based sequence model." In: *Nature methods* 12.10 (2015), pp. 931–934.

[175] Nadav Brandes, Grant Goldman, Charlotte H Wang, Chun Jimmie Ye, and Vasilis Ntranos. "Genome-wide prediction of disease variant effects with a deep protein language model." In: *Nature Genetics* (2023), pp. 1–11.

[176] Philipp Rentzsch, Max Schubach, Jay Shendure, and Martin Kircher. "CADD-Splice—improving genome-wide variant effect prediction using deep learning-derived splice scores." In: *Genome medicine* 13.1 (2021), pp. 1–12.

[177] Christopher Y Park, Jian Zhou, Aaron K Wong, Kathleen M Chen, Chandra L Theesfeld, Robert B Darnell, and Olga G Troyanskaya. "Genome-wide landscape of RNA-binding protein target site dysregulation reveals a major impact on psychiatric disorder risk." In: *Nature genetics* 53.2 (2021), pp. 166–173.

[178] Peter D Stenson, Edward V Ball, Matthew Mort, Andrew D Phillips, Jacqueline A Shiel, Nick ST Thomas, Shaun Abeysinghe, Michael Krawczak, and David N Cooper. "Human gene mutation database (HGMD®): 2003 update." In: *Human mutation* 21.6 (2003), pp. 577–581.

[179] Melissa J Landrum, Jennifer M Lee, Mark Benson, Garth R Brown, Chen Chao, Shanmuga Chitipiralla, Baoshan Gu, Jennifer Hart, Douglas Hoffman, Wonhee Jang, et al. "ClinVar: improving access to variant interpretations and supporting evidence." In: *Nucleic acids research* 46.D1 (2018), pp. D1062–D1067.

[180] Johannes Linder, Divyanshi Srivastava, Han Yuan, Vikram Agarwal, and David R Kelley. "Predicting RNA-seq coverage from DNA sequence as a unifying model of gene regulation." In: *bioRxiv* (2023), pp. 2023–08.

[181] Felix Mölder, Kim Philipp Jablonski, Brice Letcher, Michael B Hall, Christopher H Tomkins-Tinch, Vanessa Sochat, Jan Forster, Soohyun Lee, Sven O Twardziok, Alexander Kanitz, et al. "Sustainable data analysis with Snakemake." In: *F1000Research* 10 (2021).

[182] *Anaconda Software Distribution*. Version Vers. 2-2.4.0. 2020. URL: https://docs.anaconda.com/.

[183] Laura De Conti, Marco Baralle, and Emanuele Buratti. "Neurodegeneration and RNA-binding proteins." In: *Wiley Interdisciplinary Reviews: RNA* 8.2 (2017), e1394.

[184] Yiran Zhu, Liyuan Zhu, Xian Wang, and Hongchuan Jin. "RNA-based therapeutics: an overview and prospectus." In: *Cell death & disease* 13.7 (2022), p. 644.

[185] Iz Beltagy, Matthew E Peters, and Arman Cohan. "Longformer: The long-document transformer." In: *arXiv preprint arXiv:2004.05150* (2020).

[186] Zichao Yan, William L Hamilton, and Mathieu Blanchette. "Graph neural representational learning of RNA secondary structures for predicting RNA-protein interactions." In: *Bioinformatics* 36.Supplement_1 (2020), pp. i276–i284.

[187] *Flow.bio*. URL: https://flow.bio/ (visited on 09/30/2023).

[188] Paolo Di Tommaso, Maria Chatzou, Evan W Floden, Pablo Prieto Barja, Emilio Palumbo, and Cedric Notredame. "Nextflow enables reproducible computational workflows." In: *Nature biotechnology* 35.4 (2017), pp. 316–319.

[189] Žiga Avsec, Roman Kreuzhuber, Johnny Israeli, Nancy Xu, Jun Cheng, Avanti Shrikumar, Abhimanyu Banerjee, Daniel S Kim, Lara Urban, Anshul Kundaje, et al. "Kipoi: accelerating the community exchange and reuse of predictive models for genomics." In: *BioRxiv* (2018), p. 375345.

[190] Avanti Shrikumar, Katherine Tian, Žiga Avsec, Anna Shcherbina, Abhimanyu Banerjee, Mahfuza Sharmin, Surag Nair, and Anshul Kundaje. "Technical note on transcription factor motif discovery from importance scores (TF-MoDISco) version 0.5. 6.5." In: *arXiv preprint arXiv:1811.00416* (2018).

[191] Johannes Linder, Alyssa La Fleur, Zibo Chen, Ajasja Ljubetič, David Baker, Sreeram Kannan, and Georg Seelig. "Interpreting neural networks for biological sequences by learning stochastic masks." In: *Nature machine intelligence* 4.1 (2022), pp. 41–54.

[192] John Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žídek, Anna Potapenko, et al. "Highly accurate protein structure prediction with AlphaFold." In: *Nature* 596.7873 (2021), pp. 583–589.

[193] Ahmed Elnaggar, Michael Heinzinger, Christian Dallago, Ghalia Rehawi, Yu Wang, Llion Jones, Tom Gibbs, Tamas Feher, Christoph Angerer, Martin Steinegger, et al. "Prottrans: Toward understanding the language of life through self-supervised learning." In: *IEEE transactions on pattern analysis and machine intelligence* 44.10 (2021), pp. 7112–7127.

[194] Alexander Rives, Joshua Meier, Tom Sercu, Siddharth Goyal, Zeming Lin, Jason Liu, Demi Guo, Myle Ott, C Lawrence Zitnick, Jerry Ma, et al. "Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences." In: *Proceedings of the National Academy of Sciences* 118.15 (2021), e2016239118.

[195] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. "Language models are few-shot learners." In: *Advances in neural information processing systems* 33 (2020), pp. 1877–1901.

[196] Kristopher W Brannan, Isaac A Chaim, Ryan J Marina, Brian A Yee, Eric R Kofman, Daniel A Lorenz, Pratibha Jagannatha, Kevin D Dong, Assael A Madrigal, Jason G Underwood, et al. "Robust single-cell discovery of RNA targets of RNA-binding proteins and ribosomes." In: *Nature methods* 18.5 (2021), pp. 507–519.