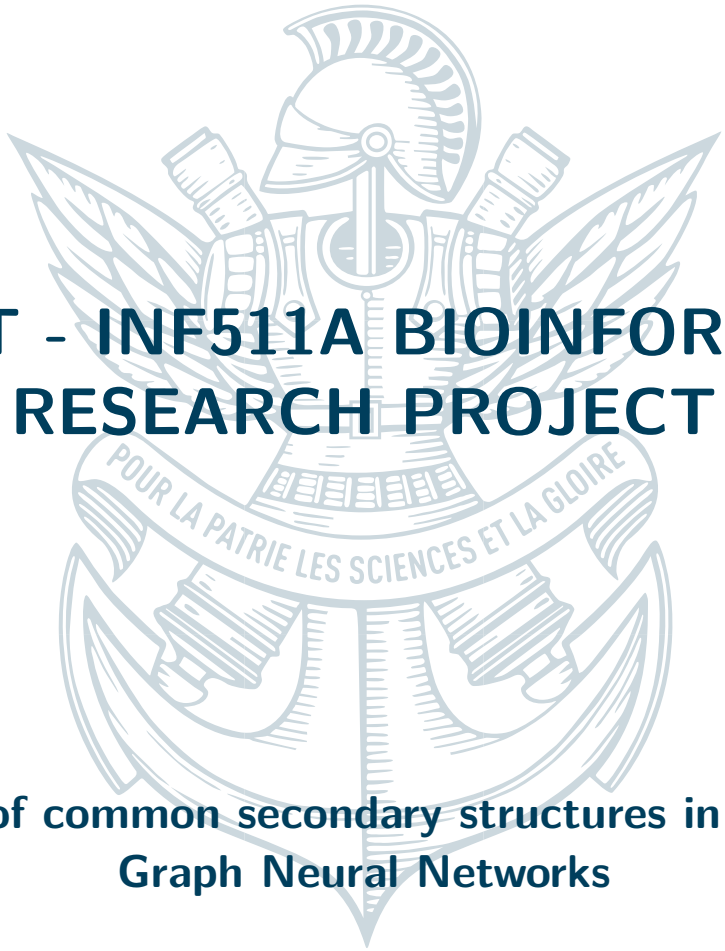

REPORT - INF511A BIOINFORMATICS RESEARCH PROJECT

Detection of common secondary structures in RNA using Graph Neural Networks

September 2023 - March 2024

Thomas Loux

CONTENTS

ACKNOWLEDGEMENTS

I would like to thank my mentors, Yann Ponty, Sebastian Will and Johannes Lutzeyer. It is always a pleasure to work with you and discuss the project every week.

1 EXPLORING THE DYNAMICS OF RNA: INSTABILITY, STRUCTURAL COMPLEXITY, AND FUNCTIONAL TASKS

1.1 Instability of RNA

RNA is a nucleic acid present in all living organisms. For a long time, this structure was considered a simple intermediary in the formation of proteins from DNA. While this function is indeed found in messenger RNAs, the discovery of non-coding RNAs has opened the door to new RNA functions.

It has been established that RNA can play a role in regulating gene expression and chromosome structure. These functions are carried out by non-coding RNAs, which are not translated into proteins. These RNAs are then called non-coding RNAs (ncRNAs).

These ncRNAs are molecules with a three-dimensional structure that can interact with other molecules. These interactions form the basis of gene expression regulation and chromosome structure regulation.

However, one should not perceive RNA as a molecule with a role similar to DNA simply because they are both nucleic acids. Indeed, RNA is characterized by being much more unstable than DNA. While this instability poses a challenge in long-term information storage, as achieved by DNA, it allows for fine regulation of RNA expression. This enables the control of the quantity of RNA present and the degree of interaction affinity. Unlike DNA, RNA cannot be defined by its most stable structure, meaning the structure that minimizes free energy. In fact, this structure is not necessarily the most prevalent in the cell. RNA can be defined as a set of structures following a Boltzmann distribution based on associated free energy [5].

$$P(\mathbf{S}) = \frac{e^{-\beta \Delta G(\mathbf{S})}}{Z} \tag{1}$$

with $\mathbf{S}$ the secondary structure, β the thermodynamic beta, the inverse of $k_B T$, $\Delta G(\mathbf{S})$ the free energy of the structure $\mathbf{S}$ and Z the partition function. The partition function is the sum of Boltzmann factors over all possible structures.

A interesting set of ncRNAs are riboswitches [4] that are able to change their structure in response to a ligand. This change of structure can be used to regulate gene expression. The study of these molecules is particularly interesting as it can be used to design new RNA-based tools.

These characteristics clearly distinguish RNA from proteins, functional molecules that have been more extensively studied using an algorithmic approach. Among these approaches, the use of neural

networks has been particularly successful, especially through the use of Transformer as for Alphafold [6]. The use of neural networks for RNA is more recent, but has already shown promising results. In particular, the use of graph neural networks has been shown to be effective with RNA secondary structures . This is the approach we will take in this project.

1.2 The structure provides the function

The function of an RNA is determined by its structure. This structure is defined by the sequence of nucleotides that compose it. These nucleotides are characterized by their base, which can be adenine (A), cytosine (C), guanine (G) or uracil (U).
The sequence of nucleotides is written as a string of letters, each letter corresponding to a base. A base pair is formed between two nucleotides if the bases of these nucleotides are complementary. The complementarity of two bases is defined by the fact that they can form a hydrogen bond. The bases A and U are complementary, as are the bases C and G. The **A-U** and **C-G** bonds are called the Watson-Crick bonds. Additionally in RNA, the **G-U** bond is also possible.
In the context of the structure, it is expected that **the base pair bonds do not intersect or cross**. This assumption is base on the observation of structures. Generally speaking, the RNA will adopt some knot structures. This hypothesis on the secondary structure allows efficient algorithmic approach using dynamic programming (RNAFold [5]). Yet other consideration are possible, for instance some pseudo-knots. Then, the problem is NP-hard [1]. Hence, we are going to make this assumption in this project.

In the case of RNAs, their structures play a crucial role in facilitating their functions. Despite mutations occurring in the sequence, the overall structure tends to remain highly conserved, underscoring the importance of structural integrity in RNA functionality. One example of an RNA with a mutation in its sequence but a conserved structure is the ribosomal RNA (rRNA). Ribosomal RNA plays a fundamental role in protein synthesis by providing the scaffold upon which ribosomal proteins assemble and catalyze peptide bond formation. Despite the essential nature of rRNA, mutations can occur within its sequence due to various factors such as DNA replication errors or exposure to mutagens. However, the overall structure of rRNA tends to be highly conserved across species, particularly within functional domains critical for ribosome assembly and function

This structure can be represented as a graph. The nodes of the graph are the nucleotides of the sequence. The edges of the graph are the base pairs of the structure, strong and weak edges are represented.

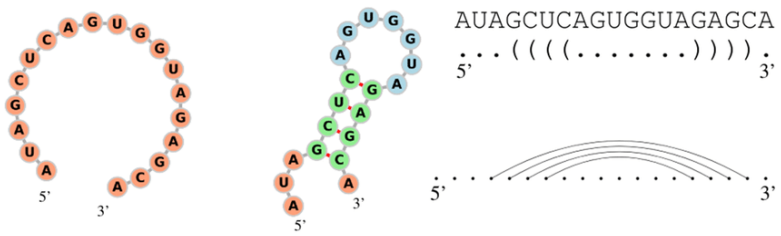

Figure 1: RNA secondary struture and link with the dot-bracket format, from [8]

2
STRATEGY TO DETECT OF COMMON SECONDARY STRUCTURES

In this project, we will focus on RNA families. These families are sets of RNAs that share similarities. Generally, the mutation on RNAs may imply deletion or addition of nucleotide. In the following, we will only consider RNAs with the same length.
The idea is that stable alternative secondary structures are conserved in the family. It should correspond to a local minimum of the free energy. Yet, around a local minima, the difference in term of structure and set of bonds between nucleotide should be small. Hence the base pair distance should be small. This implies that if we cluster the secondary structures of the family with the base pair distance, we should find clusters that correspond to the different stable secondary structures. Indeed, these alternative structures should minimized the total distance to other structures in the same cluster.

2.1 Data representation

We first start from a set of RNA sequences, extracted from a already known family of RNAs (RFAM for instance) or obtained by experimental data. We then generate a great number of suboptimal secondary structure for each sequence . The number of generated secondary structures ranges from 1000 to 100 000 samples per sequence. We generate these structures using RNAsubopt from the RNA Vienna package [5] This command uses a stochastic sampling approach to generate suboptimal structures. We also get the energy of each structure and the probability that can be derived using the Boltzmann distribution. The temperature is a hyperparameter of the model. The choice will depend on the task. For instance using a standard temperature (aroud 37°C) ensures that the model learns the most probable structures. On the opposite, if we want a model that can capture a broader landscape of structures, we can increase the temprature to sample more diverse structures. 70°C will typically be chosen in this case.
This hyperparameter is generally betaScale where:

$$\beta_0 = \frac{1}{k_B T_0} betaScale = \frac{k_B T}{\beta_0} \tag{2}$$

where T_0 is 37°C and k_B is the Boltzmann constant.
Then I typically use betaScale in the range of 1 to 2.

2.2 Graph Neural Network

We use a Graph Neural Network (GNN) to learn a representation of the graph. We use a GNN with a message passing approach. The message passing approach is based on the propagation of information between the nodes of the graph. The information is propagated through the edges of the graph. The information is propagated in the form of messages. The messages are computed from the features of

the neighboring nodes. The messages are then aggregated at the target node. The aggregation is done by summing the messages.
The current model uses Residual Gated Graph ConvNets layers [2]. The model is implemented using PyTorch Geometric [3]. The exact model is the following:

$$\mathbf{x}_i^{n+1} = \mathbf{W}_1\mathbf{x}_i^n + \sum_{j\in\mathcal{N}(i)} \eta_{i,j} \odot \mathbf{W}_2\mathbf{x}_j^n$$

where the gate $\eta_{i,j}$ is defined as

$$\eta_{i,j} = \sigma(\mathbf{W}_3\mathbf{x}_i^n + \mathbf{W}_4\mathbf{x}_j^n)$$

$\mathbf{x}_i^n$ is the representation of the node i at the layer n. $\mathbf{W}_1$, $\mathbf{W}_2$, $\mathbf{W}_3$ and $\mathbf{W}_4$ are the weights of the model. $\mathcal{N}(i)$ is the set of neighbors of the node i. $\odot$ is the element-wise product. σ is the sigmoid function.
The graph embedding is obtained by summing the representation of the nodes and passing it through a MLP.

$$\mathbf{X} = \sum_{i=1}^{N} \mathbf{x}_i$$

$$\mathbf{X} = \sigma(\mathbf{W}_6\sigma(\mathbf{W}_5\mathbf{X}))$$

The idea is to take information from the previous state of node and aggregate the information from the neighbor with a gate. These weights depend on both the source and target node. The model is then able to learn the importance of the information from the neighbor.

The hidden dimension size of the representation is a hyperparameter of the model. We set the hidden size to be the same as the final representation size. Experiments show that a higher hidden size improved the performance of the model. The hidden size is then set to 256.

The number of layers is also a hyperparameter of the model. We set the number of layers to 3. The performance of the model is not improved by adding more layers.

This model provides nodes representation. It is then possible to use a pooling layer to obtain a graph representation. The pooling layer can be a sum or mean of the nodes representation. $\mathbf{X}_{pool} = \frac{1}{N}\sum_{i=1}^{N}\mathbf{X}_i$ with N the number of nodes. This vector is then passed through a multilayer perceptron (MLP) to obtain a more accurate graph representation. The MLP is a fully connected neural network.

2.3 Training using couple of structures

The task is to make the euclidean distance of the embedding of two structures as close as possible to the actual value of the base pair distance.

$$\mathcal{L}_{distanceLoss} = (\Delta(E_1, E_2) - ||\mathbf{X}_{1,pool} - \mathbf{X}_{2,pool}||)^2 \tag{3}$$

The chosen loss is the mean square root. Yet, any regression loss could be considered such as the mean absolute error. The loss is then minimized using the Adam optimizer. The learning rate is

chosen to be 0.01.

For the training, each epochs had the same number of couples than the dataset. The couples were generated randomly at each epoch.

Regarding the number of epochs, as the dataset is built from a sampling of the Boltzmann distribution, the relevant variable to measure the training is the total number of couples seen, independently of the number of samples in the dataset. It is essentially $n_{epochs} \times n_{couplePerEpoch}$ and we choose $n_{couplePerEpoch}$ equal to the dataset size. For 100 000 samples, we fix the number of epochs to 100. This training takes around 3 hours.

2.4 Clustering

The challenge for the clustering is to choose a relevant clustering. For the purpose of the tasks, we only need to have the labels the clusters, whose number is derived from the algorithm or a hyperparameter. The clustering algorithm should be fast enough to be used in a pipeline. Usually the problem is the quadratic complexity of the clustering algorithm. Yet, we can use a linear complexity algorithm if the number of clusters is fixed. K-Means is a good candidate for this task. This issue is that normally, we would use a distance such as the base pair distance, which needs to be computed on all pairs of structures. This is a quadratic complexity. Yet, we can use the representation of the graph to compute a vector representation of the graph. Then the complexity of K-Means is linear. For now, we are going to use K-Means.

2.5 Choosing the representative

At the first glance, the representative of the cluster should be the structure that minimize the distance to the other structures in the cluster. Yet, if one uses this strategy, the obtained representatives are not the best. The possible reason is that the redundancy of the structures are not enough to ensure that this representative will indeed be the local minimum.
In order to make the representative more relevant, we find the maximum of the expected accuracy of the set of structures for each cluster. The expected accuracy is the sum of the probability of the structures in the cluster:

$$EA(S) = 2 * \gamma \sum_{(i,j) \in S} P_{i,j} - \sum_{iunpaired} P_i^u \tag{4}$$

The $P_{i,j}$ is the probability of the base pair (i, j) and P_i^u is the probability of the nucleotide i to be unpaired. The γ is a hyperparameter of the model. It corresponds to the importance rate of a nucleotide to be bounded and not unpaired.
One can obtained the maximum of the expected accuracy by using a dynamic programming algorithm. In fact, the expected accuracy can be computed for all the substructures. This is a consequence of the non crossing assumption. This strategy allows to use all the information from a set of structures. It has been used for riboswitches and has shown to be effective [7].

2.6 Choosing the relevant clusters

The number of clusters is a hyperparameter of the model. It is chosen by the user. Yet, we may have some clusters with low information about the family. Indeed, some clusters can be small or have only secondary structures from the same sequence. We can use the entropy of the clusters to choose the relevant clusters. This entropy is computed on the proportion of each sequence in the cluster. The entropy is then used to filter the clusters. The clusters with a low entropy are not relevant.

3 RESULTS

3.1 Estimation of the base pair distance

The network is able to learn the base pair distance. We can have a look at the correlation graph between the predicted distance and the actual distance. This allows to see if the model is able to capture the distance.

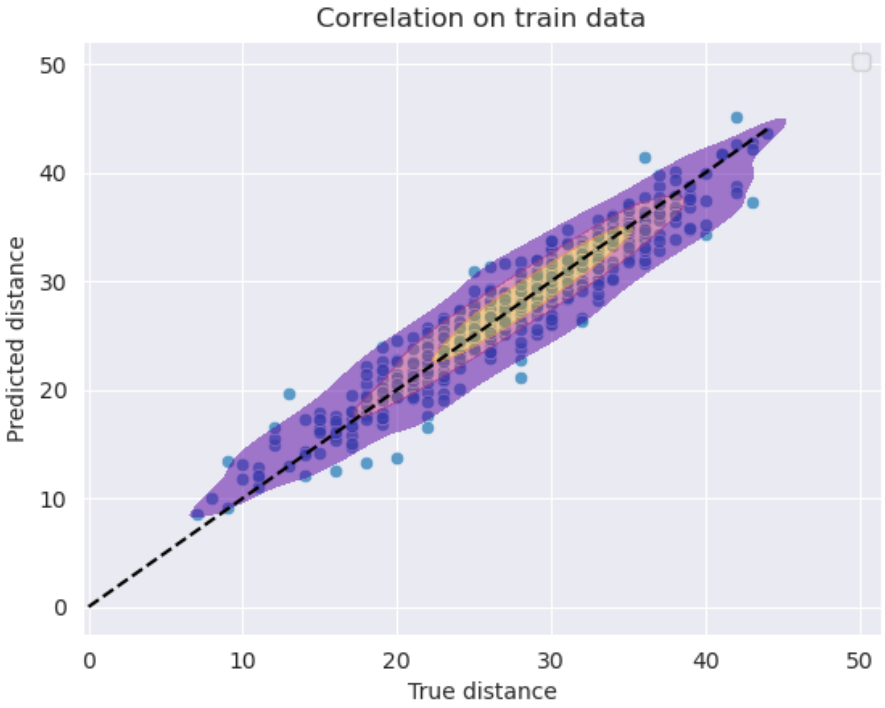

Figure 2: Correlation between the predicted distance and the actual distance

The Mean Square Error is around 3. This is a rather good result. Additionnaly the figure shows the repartition of the data points with a Kernel Density Estimate. Each colour corresponds to a third

of the data.

3.2 Alternative secondary structures

I made the experiments with the following family:

- X54124.1/910-981: GACUGCUUGGCGCAAUGGUAGCGCGUUCGACUCCAGAUCGAAAGGUUGGGCGUUCGAUCCGCUCAGUGGUCA,
- EU255777.1/1590-1519: UGGGGCGUGGCCAAGUGGUAAGGCAACGGGUUUUGGUCCCGCUAUUCGGAGGUUCGAAUCCUUCCGUCCCAG,
- CP000660.1/704452-704523: GGGCCGGUAGUCUAGCGGAAGGAUGCCCGCCUCGCGCGCGGGAGAUCCCGGGUUCGAAUCCCGGCCGGUCCA,
- X02173.1/522-593: GGAUCCAUAGCUUAAUAGUAAAGUCCUAUUUUGUCAUAAUAGAGGAUGUCAGUGCAAAUCUGAUUGGAUUCG,
- X54552.1/66-137: GUGAUUGUAAUUCAAUGGUAGAAUGCUUAUUUGUGGCAUAAGAAGUUCUUGGUUCGAUUCCAAGUAAUCACC,
- D12694.1/2533-2604: GCUUUUAAAGGAAAAGAGCCCUCCACUGGUCUUAGGCGCCAGCAUCUCUUGGUGCAAGUCCAAGUAAAAGCU,
- X16888.1/864-793: UGAGUUGUAGCCUAAUGGAAAGGCGUUUGGCCGUUAACUAAAAGAGAGCAAGAUCAAUACUUGUCGACUCAG,
- X03676.1/445-516: GCGGAAAUAGCUUAAUGGUAGAGCAUAGCCUUGCCAAGGCUGAGGUUGAGGGUUCAAGUCCCUCCUUCCGCU

This family is a set of tRNA. The tRNA is a molecule that is used to transport the amino acid to the ribosome. The tRNA has a cloverleaf structure.
We select 15 clusters and use gamma equals 3.
Using the strategy described, we can obtained the following representatives:

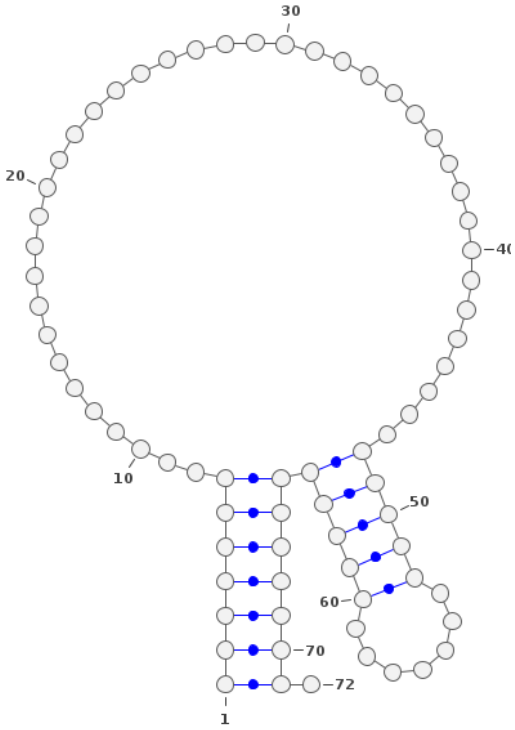

Figure 3: 1 - Representative of the clusters

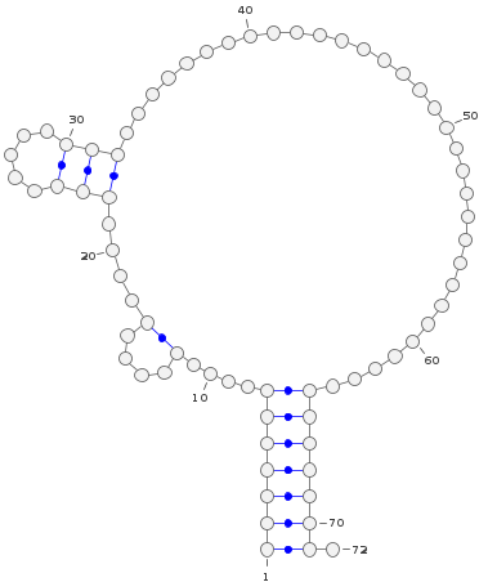

Figure 4: 2 - Representative of the clusters

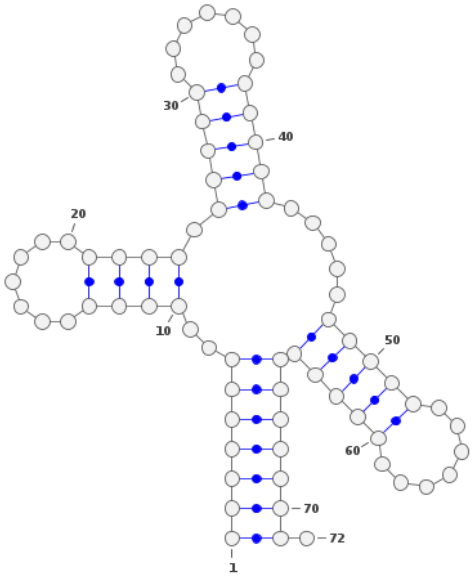

Figure 5: 3 - Representative of the clusters

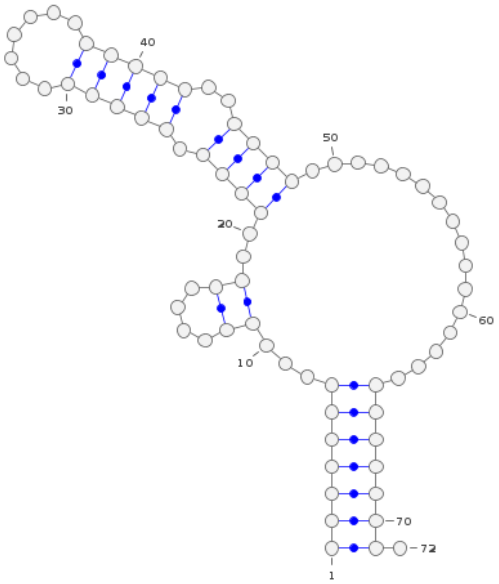

Figure 6: 4 - Representative of the clusters

The third representative is the most relevant. It corresponds to the cloverleaf structure of the tRNA. The other representatives do not seem really relevant.

4 DISCUSSION

Despite the promissing approach, the results are not yet satisfying. The main issue is the choise of the representative. The low redundancy makes it difficult to find the relevant representative. There are two ways to increase the redundancy: Increase the number of sample or decrease the BetaScale. We cannot always lower the BetaScale as it may remove the overlapping of the secondary structures among sequences. The number of samples cannot be increased indefinitely as well.
The issue is also to have families to easily evaluate the performance of the model.

5 CONCLUSION

We suggest a strategy to detect common secondary structures in RNA. The strategy is based on the use of a Graph Neural Network to learn a representation of the secondary structure. The representation is then used to cluster the secondary structures. The clusters are then used to find the relevant representatives. The representatives are then used to find the alternative secondary structures.
Although the results are still mitigated, it involves multiples tools and strategies from different contexts. Despite some difficulties to find a solution of the current problem, I have been able to get the hang of the different tools and strategies.

REFERENCES

[1] Édouard Bonnet, Paweł Rzążewski, and Florian Sikora. Designing rna secondary structures is hard. *Journal of Computational Biology*, 27(3):302–316, 2020. PMID: 32160034.

[2] Xavier Bresson and Thomas Laurent. Residual gated graph convnets, 2018.

[3] Matthias Fey and Jan E. Lenssen. Fast graph representation learning with PyTorch Geometric. 2019.

[4] Andrew D Garst, Andrea L Edwards, and Robert T Batey. Riboswitches: structures and mechanisms. *Cold Spring Harbor perspectives in biology*, 3(6):a003533, 2011.

[5] Andreas R Gruber, Ronny Lorenz, Stephan H Bernhart, Richard Neuböck, and Ivo L Hofacker. The vienna rna websuite. *Nucleic acids research*, 36(suppl_2):W70–W74, 2008.

[6] John Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žídek, Anna Potapenko, Alex Bridgland, Clemens Meyer, Simon A. A. Kohl, Andrew J. Ballard, Andrew Cowie, Bernardino Romera-Paredes, Stanislav Nikolov, Rishub Jain, Jonas Adler, Trevor Back, Stig Petersen, David Reiman, Ellen Clancy, Michal Zielinski, Martin Steinegger, Michalina Pacholska, Tamas Berghammer, Sebastian Bodenstein, David Silver, Oriol Vinyals, Andrew W. Senior, Koray Kavukcuoglu, Pushmeet Kohli, and Demis Hassabis. Highly accurate protein structure prediction with alphafold. *Nature*, 596(7873):583–589, Aug 2021.

[7] Feng Lou and Peter Clote. Maximum expected accurate structural neighbors of an rna secondary structure. In *2011 IEEE 1st International Conference on Computational Advances in Bio and Medical Sciences (ICCABS)*, pages 123–128. IEEE, 2011.

[8] Daniele Marchei and Emanuela Merelli. Rna secondary structure factorization in prime tangles. *BMC Bioinformatics*, 23, 08 2022.