Master Thesis

Secondary structure motif determination in ncRNA via graph kernel based computational models

Kiran Kumar Telukunta

February 2012

Albert-Ludwigs-Universität Freiburg im Breisgau
Technische Fakultät
Institut für Informatik

Thesis Period

August 1, 2011 – February 29, 2012

Gutachter

Prof. Dr. Rolf Backofen
Prof. Dr. Martin Riedmiller

Betreuer

Dr. Fabrizio Costa

Dedication

Erklärung

Hiermit erkläre ich, dass ich diese Abschlussarbeit selbständig verfasst habe, keine anderen als die angegebenen Quellen/Hilfsmittel verwendet habe und alle Stellen, die wörtlich oder sinngemäß aus veröffentlichten Schriften entnommen wurden, als solche kenntlich gemacht habe. Darüber hinaus erkläre ich, dass diese Abschlussarbeit nicht, auch nicht auszugsweise, bereits für eine andere Prüfung angefertigt wurde.

Table of contents

Zusammenfassung

ncRNA ist ein funktionelles Molekül, das noch nicht in ein Protein übersetzt wurde. In letzer Zeit hat ncRNA im Fachbereich BioInformatik deutlich an Bedeutung gewonnen, z.B. in der Therapeutik, Chemoinformatik und vielen anderen Bereichen der Biologie.

Die Nukleotid-Zusammensetzung und ihre Struktur (Identität der gepaarten und ungepaarten Nukleotide) bestimmen die Funktion der ncRNA und ihre Eigenschaften. Wichtige analytische wissenschaftliche Werkzeuge, wie Sequenzalignment und Clustering Algorithmen, basieren auf energetischen Überlegungen, um spezifische Anfragen genau zu beantworten. In der Realität machen diese Algorithmen Fehler, wenn ihre Annahmen (z.B. Energie-Additivität) verletzt werden, da sie insbesondere nicht-lineare Effekte nicht berücksichtigen

Um diese Eventualitäten zu überwinden, kann man Fragen stellen in Bezug auf nicht-lineare funktionale Abhängigkeiten, die aus bekannten Beispielen gelernt werden können (oder Teilen von Beispielen) oder von Maßen aus verschiedenen suboptimalen RNA-Struktur-Vorhersagen. Angesichts der Bedeutung der strukturellen Elemente in ncRNAs sollten diese Verfahren idealerweise in der Lage sein in strukturierten Domänen zu arbeiten, d.h. Graphen als Eingabe zu akzeptieren. Diese Methoden werden zur Familie der Kernel-Maschinen gehören, da es diese Klasse von Algorithmen ermöglicht, heterogene Funktionen zu nutzen und komplexe Datenstrukturen wie Sequenzen von Graphen als Eingabe zu akzeptieren.

Das Ziel der Arbeit ist es, Berechnungsmodelle zu entwickeln, die die Identifizierung von Sub-Graphen innerhalb des ncRNA Faltungsgraphen ermöglichen, die charakteristisch für die Entwicklung biologischer Funktionen sind; weiterhin die Entwicklung von Kernel-Modellen, um die *RNA Sekundärstruktur* und ihre Vorhersage in Bezug auf die Genauigkeit zu verbessern.

Abstract

ncRNA which is a functional molecule but yet not translated into protein has significantly taken importance in the field of bioinformatics, therapeutics chemoinformatics and for the advancement of science.

The nucleotide composition and its structure (identity of paired and unpaired nucleotides) determine the function of ncRNA. Key analytical tools such as folding, alignment and clustering algorithms rely on energetic considerations to generate the accurate response to specific queries as they are designed. In reality, these algorithms become inaccurate while considering the non-linear effects with underlying assumptions (energy additivity), when violated.

To overcome these eventualities, one can formulate key parameters in terms of non-linear functional dependencies that can be learned from known examples (or parts of examples) or from suboptimal *RNA* structure prediction. Given the importance of the structural element in ncRNA these methods should ideally be able to work in structured domains i.e. they should be able to accept input graph data structures. The methods will belong to the family of kernel machines, since this class of algorithms allows to use heterogeneous features and to accept complex instances such as sequences of graphs as input.

The aim of the thesis is to develop computation model capable of identifying subgraphs within the ncRNA folding graph that are characteristic of biological functions. Further subject them to kernel models to improve the *RNA secondary structure* and its prediction in terms of accuracy.

1 Introduction

1.1 Motivation

In bioinformatics *RNA* has taken significant role in study of Genes and their structures. *RNA* which is present in the form of sequences of long chain of nucleotides. The *RNA* plays the major role in biological reactions in dictating Gene Expressions leading its significant importance in the field of bioinformatics.

The main decree in bio-informatics says that information flow happens from *DNA* to *RNA*. There are numerous applications, research interests on *DNA* [Metzker, 2005]. Ribonucleic acid (*RNA*) is biologically important type molecule that consists of long chain of nucleotides. Each nucleotide unit consists of nitrogenous base, and a phosphate. The sequence of nucleotides allows *RNA* to encode genetic information. In many cases, *RNA* is similar to one of the double stranded *DNA* sequence. *RNA* is determined by the sequence of *DNA* in several cases. *RNA* is used in turn, is used to direct production of protein.

Nucleotides in *RNA* consists of ribose sugar with carbons numbered $1'$ through $5'$. A base is attached to the $1'$ position. Generally adenine(A), cytosine(C), guanine(G) or uracil(U) in which adenine and guanine are purines, cytosine and uracil are pyrimidines.

Initially most of the research happened in the direction of *DNA* but in recent years it is found that *RNA* is the major element of information in genomics. The number of protein-coding genes are in greater magnitude in human or mice [Taft et al., 2007]. The high percentage of 98% of human genome are non-protein-coding among which are transcribed into short and long non-coding *RNA*'s (ncRNAs) [Taft et al., 2010]. Since then *ncRNA* became significant in the field of medical field and got immense focus on the fundamental mechanisms by which ncRNAs facilitate normal development consecutively to abate in curing diseases. The potential use in therapeutic targets.

Hence the need of studying structure of *RNA* and thereby *ncRNA* vastly increased. Principally the *RNA* study started with primary structure and three-dimensional conformation characterized by various loops and twists.The tertiary structure determines the biochemical activity of *RNA* molecule through X-ray diffraction or biochemical probes which are extremely costly and time consuming techniques and mostly are insufficient to study structure. Gradually this has been simplified and constraining the study only to the extent of base pairs which are involved in the sequence. These collection of base pairs when constructed as linear structure from $5'$ terminus through to $3'$ terminus a solid line is drawn between complementary strands of hydrogen bonded nucleotides which depicts the *secondary structure* (can be seen in Figure 1). After the formation of *secondary structure* upon processing tertiary structure prediction has been more easier and yielded better results [Zuker and Sankoff, 1984].

However, results obtained by various methods such as *free energy minimization* with nearest neighboring parameters [Doshi et al., 2004] where it uses window size, percent suboptimality and the inclusion or exclusion of additional energy calculations.which being a uncertain and inconsistent parameter gives rise to various suboptimal structures and it becomes difficult to find a better *secondary structure*, with comparative sequence analysis [Deigana et al., 2009] which consider only *RNA* regions which are only in higher order tertiary interactions which are tightly constrained by such interactions also lead to less effective structure prediction.

1.2 RNA Secondary structure

Major structural difference between *RNA* from *DNA* is the presence of a hydroxyl group at the 2*'* position of the *ribose sugar* in *RNA*. The pairs which are formed are **GC**, **AU** and **GU** are in both directions.

Definition 1.2.1. *RNA Secondary structure: Let $S \in \{A, C, G, U\}$ be a RNA sequence. A RNA secondary structure of S is a set of pairs P*

$$P \subseteq \{(i, j)\}|1 \leq i < j \leq n,\ S_i \ and\ S_j\ form\ a\ bond.$$

Where S_i and S_j are complementary iff
$(S_i, S_j) \in \{(G, C), (C, G), (A, U), (U, A), (G, U), (U, G)\}$

The *secondary structure* prediction can be done majorly with three techniques [Zuker and Sankoff, 1984]:

- Examine all possibilities usually with graphical procedures and better trial and error techniques.
- With the law of thermodynamics where minimum free-energy is calculated and based on which the prediction is done.
- Another approach with phylogeny, which can be used if the sequences for functionally identical molecules have been determined for several organisms or organelles. If two or more molecules have closely related primary structures or identical biological functions, the strategy is to search for a *secondary structure* common to all of them.

Among the various forms of structure prediction second approach with thermodynamics is more popular where in the *RNA* structure is often predicted from sequence by *free energy minimization.*

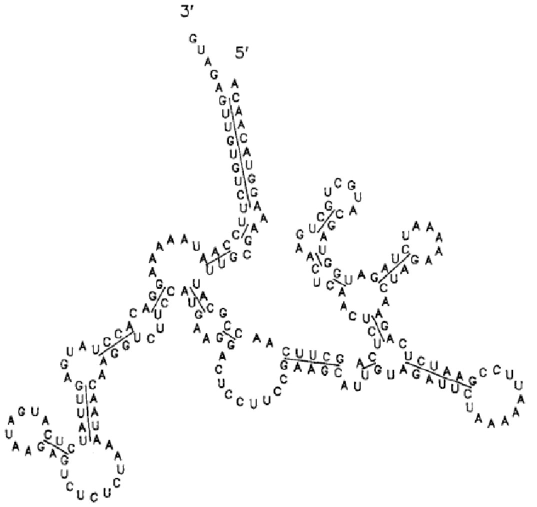


Figure 1: Secondary structure of fragment of the Cauliflower Mosaic Virus. The linear structure begins at the 5′ terminus and continues to the 3′ terminus. The solid lines are drawn between complementary strands of hydrogen bonded nucleotides. Picture taken from Zucker & Sankoff‘s article [Zuker and Sankoff, 1984]

1.3 Free energy minimization

1.3.1 Thermodynamics

Free energy minimization is the predominant model predicting *RNA secondary structure* and have various methods evolved during years. Evolution of *secondary structure* brought up many different methods and the first algorithm which was introduced more than thirty years before is based on the nearest neighbor energy model [Zuker and Stiegler, 1981]. It defined many terms based on the free energy of a structure and of its equivalent graph. A face of graph is defined to be any planar region bounded on all sides by edges. A face with a single interior edge is called a hairpin loop. Faces with two interior edges are classified into groups namely stacking region, bulge loop, interior loop and bifurcation loop. If F is a face, $E(F)$ denotes the free energy associated to it and for each face. The summation of all the faces in the structure denotes the energy of the structure. Based on this energy an algorithm is developed which selects structure with minimum *free energy among* many available structures. The conclusion is, with structures having nucleotides from 571 to 765 about 80% of the indicated base pairings survived and preserved for the final model. However when the number of nucleotides increases the preservance decreases.

Considering a case for a given function with a single sequence is known, in which case, sequence dependence of stability for the various motifs found in *RNA* is approximated. Earlier approximations are based [Mathews et al., 2004] on experiments published before 2004. However later experiments [Chen et al., 2004] have significantly revised models for approximating loop stabilities. Studies on the thermodynamics of small internal loops show that size symmetric internal loops have more sequence dependence than size asymmetric loops. For example, for internal loops closed by GC or CG pairs, stabilities of 2 x 2 nucleotide loops range from 2.2 to -2.9 kcal/mol whereas those of 1 x 3 nucleotide loops range from 3.3 to 1.6 kcal/mol

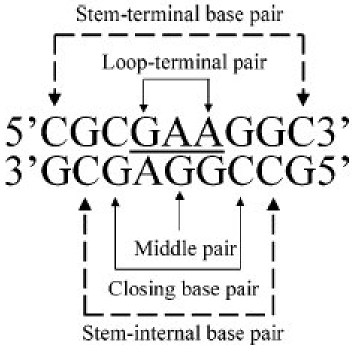


Figure 2: Schematic representation which shows the nomenclature for base pairs in duplexes with 3 x 3 internal loops.Taken from [Chen et al., 2004]

at 37 ℃. But the energy parameters taken in case of [Chen et al., 2004] for 3 x 3 internal loops as shown in Figure 2, are based on knowledge of 2 x 2 and 2 x 3 internal loops. The 3 x 3 internal loops, are the smallest size symmetric loops with a potential noncanonical base pair and the flexibility of internal loops will increase as the loop size increases. It is presumed that 3 x 3 internal loops differ from 2 x 2 loops more than from 4 x 4 and larger size symmetric loops. Thus, insights acquired from 3 x 3 loops should improve approximations for stabilities of 3 x 3 and larger internal loops.

Based on thermodynamics considering free energy minimization *RNA secondary structure* can be predicted in three different ways:

- Applying statistical mechanics of *RNA* folding based on partition function [Mathews and Turner, 2006].
- Using algorithms that allow pseudoknots.
- Finding the *secondary structure* common to set of homologous sequences.

After two decades of refined measurements of thermodynamic parameters, the problem of not reaching best accuracy exists [Doshi et al., 2004] and the predicted structures some times completely does not match the *secondary structure*s.The main reason for this situation after evolution of methods in last two decades is due to the intrinsic properties of the folding space where the grouping the structures into similar structures and the kinetics of the evolved folding with algorithms [Ding and Lawrence, 2003].

The drawback of Zucker algorithm is, heuristic approach is used during which, the redundant structures were eliminated during the process some similar structures were lost to get a better observer view, but it interrupts the probabilistic analysis.

As bioinformatics developed, a number of methods for predicting structures aroused which became significantly important for analysis in the field of bioinformatics leading many diversified solutions and probabilistic approach is gradually yielding many solutions. One of such kind of probabilistic model is *Hidden Markov Model* it evolved from *Markov process* which is a old mathematical process. The use of *Hidden Markov* *models* as the basis for profile searches to identify distant members of *RNA* sequence families, and the inference of phylogenetic trees using maximum likelihood approaches. [Durbin, 1998]. It can be formally defined as:

Definition 1.3.1. ***Hidden Markov Model**: HMM is defined as a tuple* $M = (n, m, P, A, B)$*, where* n *is the number of hidden states,* m *is the number of observable states,* P *is an* n*-dimensional vector containing initial hidden state probabilities,* A *is the* n X n*-dimensional transition matrix containing the transition probabilities such that* $A[i, j] = P(Y(t) = y_i | Y(t-1) = y_j)$ *and* B *is the* m X n*-dimensional emission matrix containing the observation probabilities such that* $B[i, j] = P(O = o_i | Y = y_j)$*.*

It tries to solve more intrigue problems finding structure as follows [De Fonzo et al., 2007]:

- **Evaluation**: To compute the probability that any model generates given sequence of observations, where *forward* and *backward* algorithms are used.
- **Decoding**: Extracting sequence of internal states that has, as a whole the highest probability and to find for each position the internal state which has the highest probability, where *Viterbi* algorithm is used.
- **Learning**: with available sequence of observations finding an appropriate model based on most probable sequences, *Viterbi learning* is extensively used for these kind of problems. Also, *Baum-Welch* algorithm used for most probable internal states. It can be more described as hypothesis where if a set of possible internal states, the set of possible external states and sequences of emissions are known. The problem is to estimate the model i.e the transition and emission probabilities. Mathematically expressed as:

  Let $E^j \equiv (e_k^j, k = 1, \cdots, L^j)$ $1 \leq j \leq R$ be the given sequences of emissions, and $S^j \equiv (s_k^j, k = 1, \cdots, L^j)$ $1 \leq j \leq R$ the associated (unknown) sequences of internal states.

The *ncRNA* has stable and physiologically relevant *secondary structure*s which are unavailable while coding *RNA*, which are palindromic tracts most of the time and it is required to recognize those palindromic sequences. In general, standard HMM is a stochastic regular grammar and not suitable to recognize palindromes. where as shown by [Yoon and Vaidynathan, 2004] based on SCFG and which are higher order relative, can be used for modeling *RNA* secondary structures and to detect *ncRNA* genes.In method [Yoon and Vaidynathan, 2004] an extension of tradition HMM context-sensitive HMM been proposed. Where some states are equipped with auxiliary memory. The data which is stored in auxiliary influences emission probabilities and the transition probabilities of the model which is termed as *context-sensitive state* $\mathbf{C}_n$. Some influences can be to adjust the emission probabilities of $\mathbf{C}_n$ such that it emits the same symbol with high probability or to generate the complementary base. A typical HMM which generates stem-loops is show in Figure 3

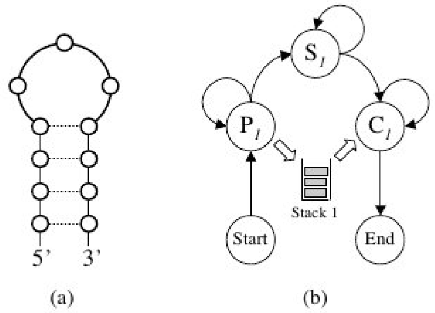


Figure 3: (a) Typical stem loop, where dotted lines indicate the interactions between bases that form complementary base-pairs. (b) An HMM which generates stem-loops. Taken from [Yoon and Vaidynathan, 2004]

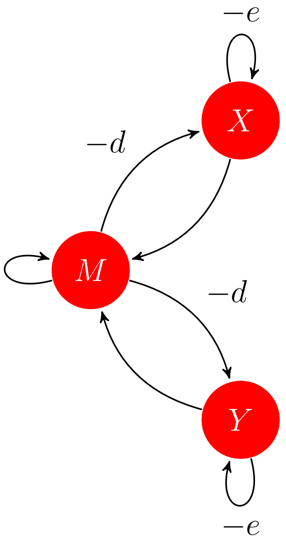


Figure 4: Finite state machine for gaps alignment.

The states in HMM are applied with *finite state automata* with multiple states, as a convenient description of more complex dynamic programming algorithms for pairwise alignment. These tools are basis for the probabilistic interpretation of the gaped alignment process, by converting them into HMMs. The approach results into a model which can be used for analyzing reliability of the alignment obtained by dynamic programming.

For any model of pairwise alignment with HMM a finite state automaton is needed in this case we require three states, M states for the matches and another two states for inserts, which are X and Y as given in Figure 4.

These states can be formally represented as follows:

$$V^M(i,j) = s(x_i, y_j) \ + \ max \begin{cases} V^M(i-1, j-1), \\ V^X(i-1, j-1), \\ V^Y(i-1, j-1); \end{cases}$$

$$V^X(i,j) = max \begin{cases} V^M(i-1, j) - d, \\ V^X(i-1, j) - e; \end{cases}$$

$$V^Y(i,j) = max \begin{cases} V^M(i, j-1) - d, \\ V^X(i, j-1) - e; \end{cases}$$

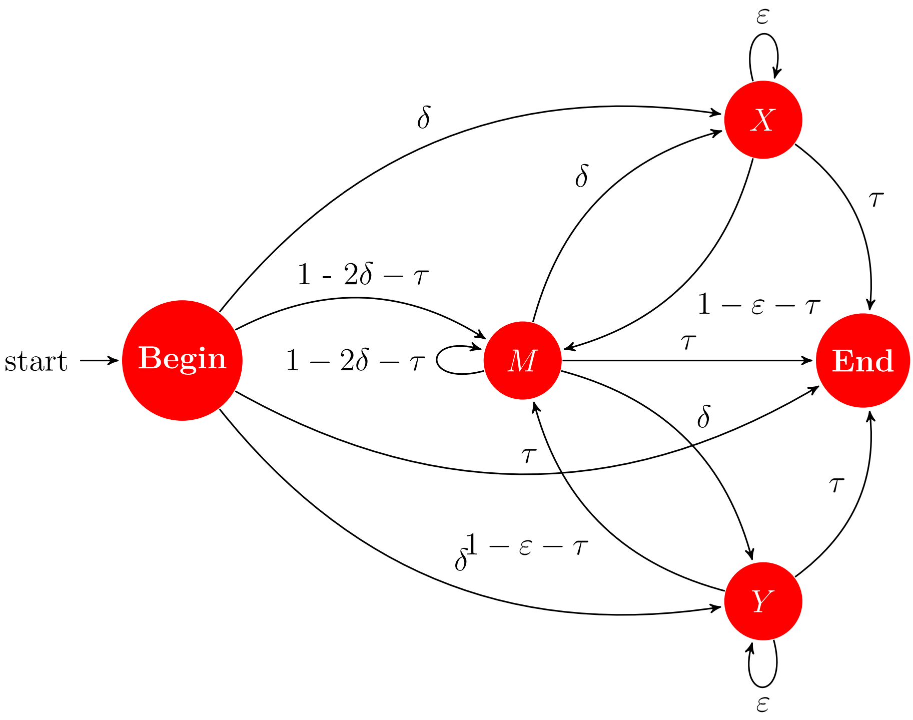


Figure 5: Probabilistic model with *Begin* and *End* states a complete model.

The finite automate in Figure 4 could be useful to the *global alignment* and to apply it to HMM it needs some changes with probabilities as shown in Figure 5, such that probabilities to be given to both emissions of symbols from the states and also for the transition states. The *transition probabilities* must oblige the requirement between the states such that for all the transitions leaving each state sum to one. To make **??** into complete model it should be added with *Begin*(initialisation) and *End*(termination) states which enables to every kind of sequence to be processed.

With initial and terminal states constitutes a formal model as shown in Figure 5. The *End* state provides the parameter for probability of transition into *End* state which is denoted by τ. According to the model required *Begin* state can be marked conveniently either similar to M or according to the requirement. Now the model satisfies the HMM, clearly it can be seen that this model emits *pairwise alignment* instead of *single sequence*. This is termed as *pair-HMM*. Based on this algorithms are formed such as *Viterbi Algorithm* to find the best alignment. A *pair-HMM* can also be formulated for local alignment.

1.4 Suboptimal Folding

Existence of several methods and wide range of algorithms give rise to range of alignments with nearly same probability and largely same kind of scoring pattern for folding is produced. As the problem of finding a tertiary structure is complex, instead a scaffold is obtained from secondary structure, which would give both ge-

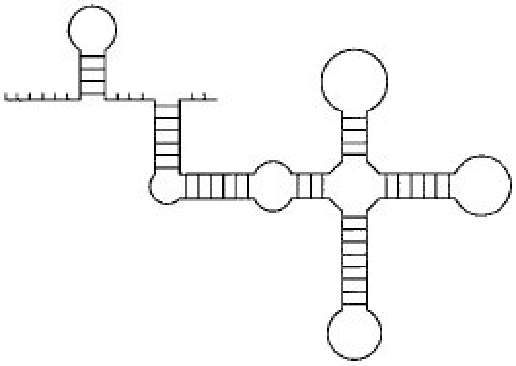


Figure 6: Base pairs are stacked and unpaired positions are free ends taken from [WUCHTY et al., 1999]

ometric and thermodynamic dimensions of the tertiary structure. In formation of *secondary structure*, *base pairs* performs major role, energy minimization and various other parameters gave rise to several probabilistic models as most of these parameters does not have certainty all these models which get to the nearly accurate structure give rise to interesting feature called *suboptimal folding.* There are various approaches in finding such foldings these can be differing in positions when compared with optimal alignment, or can be found by adjusting energy parameter which is a biologically inconsistent parameter or some other properties according to specific task with the *RNA secondary structure* such as largest number of admissible base pairs. Secondary structure can also be produced with discrete graph structures, where a *secondary structure* is first transformed into a graph structure as shown in Figure 6 and then *graph theory* knowledge is applied. In this thesis one such method can be seen in the coming sections, with improvisations on *suboptimal folding.*

Suboptimal structures obtained with various parameters are computed with the help of algorithms to improvise, Zuker's suboptimal program which utilizes the dynamic programming approach where for a sequence of length n atmost $n(n-1/2)$ suboptimal structures are produced. However, it does not address the problem of finding exhaustive suboptimal structures. This is been improved by the Wutchy's algorithm [WUCHTY et al., 1999] which generates all possible *suboptimal* folds within specified range from the *minimum free energy.* The idea of the algorithm is taken from Waterman and Byers solution [Byers and Waterman, 1984]. which is used to obtain near-optimal sequence alignments with the solution to the shortest path problem in networks.

Waterman Byers proposed the idea of *maximum matching* in which a set of edges connecting the nodes consists of two disjoint subsets after one is common set represents the covalent backbone connecting node i with node $i+1$, $i = 1, \ldots n-1$ and the other one is sequence specific set consists of a set $\mathcal{P}$ edges

$\mathcal{P} = \{i \ . \ j, \ i \neq j \ and \ j \neq i+1\}$

The above equation represents the admissible hydrogen bonds between the bases at positions i and j such that every edge in $\mathcal{P}$ connects a node to at most one other node also the pseudoknot constraint is met. Here if both $i\ .\ j$ and $k\ .\ l$ are in $\mathcal{P}$ then $i < k < j$ implies that $i < l < j$. Here the set of admissible base pairs that are considered are Watson-Crick pairs **{AU, UA, GC, CG}**$and\{GU, UG\}$. These are the same pairs which are also considered in this thesis. Here the the problem of finding the largest possible set $\mathcal{P}$ of admissible base pairs within constraints of above equation is termed as *maximum matching*.

This problem is looked as a discrete graph problem and solved by graph theory, a *matching* in an undirected graph $\mathcal{G}$ is a set of edges, where no two of which have a vertex in common. Also any set $\mathcal{P}$ of base pairs compliant with the definition of *secondary structure* is a *matching*. The dynamic programming approach to compute the maximum number of admissible base pairs in this problem can be briefly said as follows: Let $P_{i,j},\ i < j$ denotes the maximum number of base pairs on the sequence segment $[j, j]$.. $P_{i,j}$ can be defined recursively:

$P_{i,j}$ can be defined recursively:

$$P_{i,j} = \max\{P_{i,j-1}, \max_{i \leq l \leq j-2}\{(P_{i,l-1} + 1 + P_{l+1,j-1})\rho(a_i, a_j)\}\} \tag{1.1}$$

$$\rho(a_i, a_j) = \begin{cases} 1, & \text{if } a_i \text{ and } a_j \text{ can pair;} \\ 0, & \text{otherwise} \end{cases}$$

where $a_i \in \{A, U, G, C\}$ denotes the base at position i and $\rho(\cdot, \cdot)$ is an indicator function biophysical pairs.

The recursion equation Equation 1.1 works by filling the P array in such a way that all smaller fragments needed in the computation of $P_{i,j}$ have already been computed. The bases are added sequentially from the $3'$ end, and the program takes care if the added base and some position downstream improves the total number of pairs on the segment, compared to the initial status of the segment as compared to leaving the added base unpaired. After the procedure maximum number of base paire i.e $P_{max} = P_{1,n}$. A structure with P_{max} pairs is obtained by tracing back through the complete P.

1.4.0.1 Suboptimal experiences

With the inspiration from Watermann-Bayer and many other researches there were attempts to find *secondary structure* with *suboptimal foldings* we see some of them here to have understanding on the past experiences.

As dynamic algorithm is the key for Watermann-Bayer approach inspired by dynamic programming in paper Mathews et al. [2004] they incorporated chemical modification constraints into dynamic programming algorithm for prediction of RNA secondary structure. It defines *Nearest-Neighbor parameter* which is thermodynamic parameter. Which used RNASTRUCTURE tool for predicting the proposed structure. Also used *Nearest-Neighbor Parameters* for prediction of *RNA* conformational free energy at 37Ľ.

In the findings of [Doshi et al., 2004] they concentrated on 16S rRNA sequences for their dataset. The paper considered a window size(W), percent suboptimality(P), and the inclusion or exclusion of additional energy calculations based on coaxial stacking (efn2) The energy range for computed folding is established by the percent suboptimality variable. The energy range is computed as $(\Delta)_{min}$ to $\Delta G_{min} + \Delta\Delta G$, where $\Delta\Delta G$ is P of ΔG_{min}. The window size variable estimation gives the difference between the suboptimal folds by requiring that given folding has at least W *base-pairs* which are computed. After which the accuracy is calculated. For *RNA* structure prediction they used *Mfold* (3.1) was used.

The paper has defined couple of terms *RNA Contact Order* which is the average sequence separation between pairs of amino acids involved in non-covalent interactions i sdefined as *Contact Order* and the *RNA contact distance* which is separation on the RNA sequence between two nucleotides that base-pair. Any *base-pair* with *contact distance* of 100 nucleotides or less to be “short range” and *contact distance* of 100-501 nucleotides are considered to be “mid-range” and greater are termed as “long-range” and in their experiments is seen that short-range base-pairs predicted more accurately than compared with long-range sequences. Which gives an evidence that these ac curacies depend on the length of nucleotides for predicted structures.

To analyse more on *suboptimal foldings* it introduced new metrics to examine the *suboptimal* sequences on 496, 16SrRNA sequences. One the amount of variation and the $\Delta\Delta G$ difference (before evaluating with their algorithm ef2) for pairs of structure predictions in the suboptimal population. Second, how many additional unique, canonical base-pairs in comparative models were found in the suboptimal population and also monitored how many were incorrect base pairs were predicted. Last and thirdly which comparative base-pairs were predicted correctly in all, an intermediate number, or no structure predictions, ini the set of *suboptimal foldings*.

The entire 16S rRNA dataset of 496 comparative structure models contained a total of 191,994 unique canonical, comparative base-pairs. Among which 81,934 of these canonical base-pairs were predicted with Mfold 3.1 to be in a *minimum free energy* structure. For more results refer Doshi et al. [2004] which has extensive results.

Significant observations seen by the paper were when all the base-pairs in the suboptimal populationwere included in accuracy computation, it observed a 30% increase in average accuracy per sequence and other observation that there were 1,664% increase in overall number of base-pairs which were not in comparative model compared to optimal structure prediction.

It is also concluded that with newest nearest-neighbor energy values, can predict the secondary structure base pairs in comparative model structure models for different

RNAs.

The results were motivational enough to see that there can be more improvement in the accuracy when they are subjected to graph discrete structures. Which motivated to choose *kernel* graph models. One of which is [Costa and De Grave, 2010]. Which we discuss in the later sections.

2 Graph Kernel Models

Conventionally *free energy minimization* methods are approached as we have seen preceding sections to predict *RNA secondary structure.* However, with the existence of complex problems in predicting *RNA secondary structure* one The idea of graph *kernel* models came into existence which we see in the coming sections. Also there is a similar kind of method where it involves learning algorithms. These are fully automated as they more of statistical measures and analysis which try to derive a value model upon which based on policy cost is evaluated and then applied to testing phase. One of which is an interesting *RNA secondary structure* model[Do et al., 2006] which is a non physics model.

Although this is not complete graph model but it is being a non-physics model and uses SCFG.

2.1 Graph Notations

As discussed in subsection 1.3.1 to solve *secondary structure* problem they can be interpreted as discrete Graph notations we require some notations to work on such kind of model. Here in this this this model is taken from the motivation paper [Costa and De Grave, 2010]. Here the notations are followed from Gross and Yellen [2004]

Definition 2.1.1. ***Graph:*** *A graph $G = (V, E)$ consists of two sets V and E. The notion $V(G)$ and $E(G)$ is used when G is not just one graph considered. The elements of V are called vertices and the elements of E are called edges. Also the distance between two vertices is taken which is denoted as $\mathcal{D}(u, v)$. $\mathcal{D}$ is the shortest possible path between two vertices been taken.*

Definition 2.1.2. ***Neighborhood Subgraph:****The neighborhood of radius r of a vertex v is the set of vertices which is less than or equal to r from v and is denoted by $\mathcal{N}_r^v$. In a graph G, the induced-subgraph on a set of vertices $W = \{w_1, \ldots, w_k\}$ is a graph that has W as its vertex set and it contains every edge of G whose end points are in W. The neighborhood subgraph of radius r of vertex v is the subgraph induced by the neighborhood of radius r of v and is denoted by $\mathcal{N}_r^v$.*

Definition 2.1.3. ***Isomorphism:*** *Two simple graphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ are said to be isomorphic, which is denoted as $G_1 \backsimeq G_2$, if there is a*

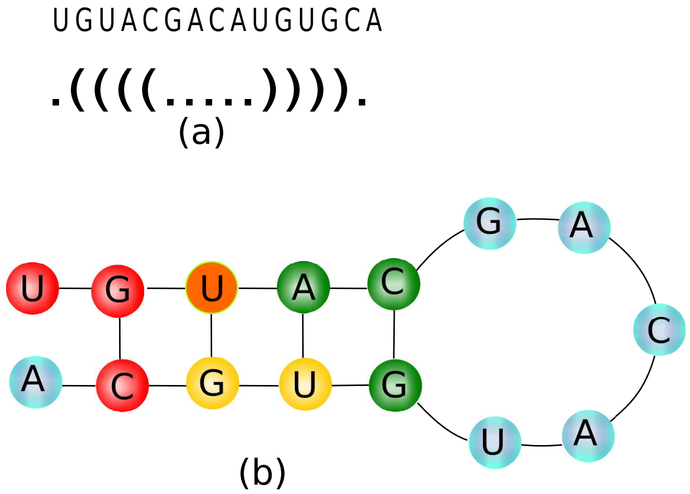


Figure 7: (a) Showing the sequence and its dp format sequence. (b) It is a basic RNA secondary structure generated from the result of RNAshapes

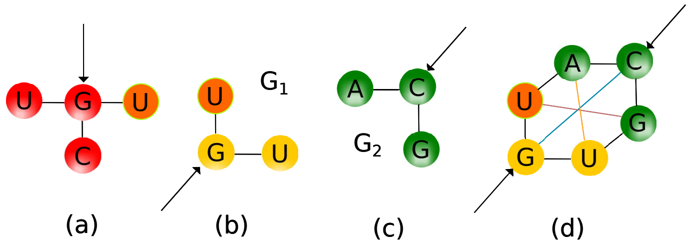


Figure 8: These are *subgraphs* notation of the structure in Figure 7 and these are rooted graphs where the arrow is pointing each subgraph is rooted at that point.

bijection $\phi : V_1 \rightarrow V_2$, such that for any two vertices $u, v \in V_1$, there is an edge uv if and only if there is an edge $\phi(u)\phi(v)$ in G_2.

With *isomorphism* structure is preserved and *bijection* is satisfied. If the label information is also preserved by the labeled graphs then they are isomorphic and denoted as $\mathcal{L}(\phi(v)) = \mathcal{L}(v)$.

When the above graph definitions are interpreted on *RNA secondary structure* it can be seen in the picture Figure 7

The Figure 7(a) is a simple *RNA Secondary structure* sequence taken from a *Fasta* format (see subsection 3.1.1) file along with its *Dot-Bracket Notation (DBN)* (see subsection 3.1.1) sequence. Figure 7(b) shows the secondary structure of the DBN. And the Figure 8 shows the *sub-graphs* of the structure in Figure 7 where one can the matching colors representing the extraction portion of the *sub-graph*.

The subgraphs Figure 8(a) and (b) are The *isomorphic* as it can be seen in Figure 8(d) where each *vertex* has a *vertex* satisfying either with base pair function or right-left node.

Definition 2.1.4. ***Isomorphism invariant****: It is a graph property that is identical for two isomorphic graphs (e.g the number of vertices and/or edges.) This can also be verified that for isomorphism is an isomorphism invariant that is identical for two raphs if and only if they are isomorphic.*

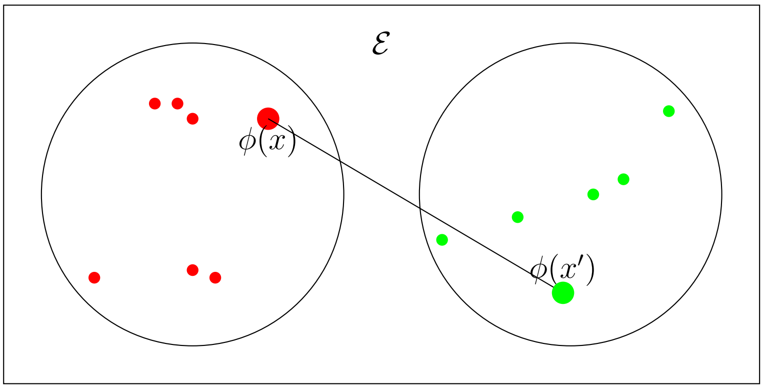


Figure 9: A simple *kernel* showing mapping of similar objects in the Euclidean space $\mathcal{E}$

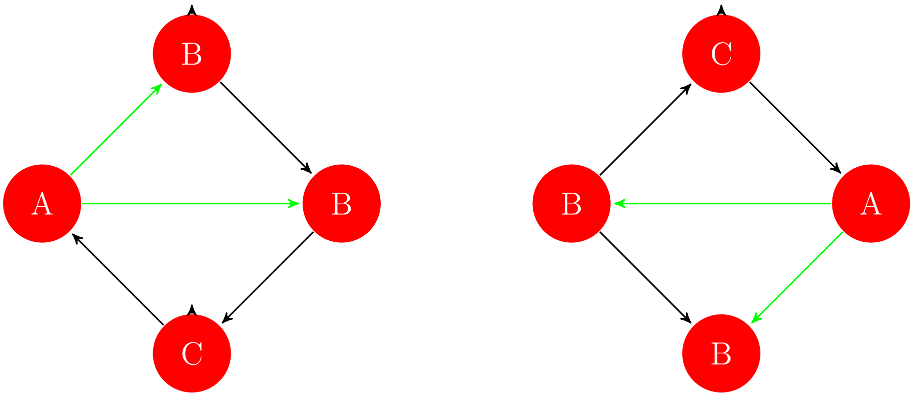


Figure 10: Pair wise comparison of substructures of directional graph

2.2 Kernels

Several problems related to statistics and pattern recognition problems available discrete structures such as trees, strings, sequences can be solved by extracting similarities and mapping them appropriately. Organize the problem into a space such that it can be formally mapped to the similarities according the features. A simple illustration as shown in the Figure 9. The key idea is to map similar objects in the Euclidean space $\mathcal{E}$. Upon formalizing the given space the measure of similarity can be shown as $K(x, x') = \langle\phi(x), \psi(x')\rangle$.

When a *kernel* extended to a pair of graphs of any two graph structures shown in Figure 10 similarities can be found in subgraph $AB - AB$ in both the graphs can be formulated as *kernel*. Similarly, defined by such discrete structures in [Haussler, 1999] and represented for various methods which has defined series representations of discrete structures using general type of *kernel* function and is termed as *convolution kernel*. For any structures an explicit formula $\{\phi_n(x)\}_{n\geq 1,}$for the inner product i.e *kernel* is formalized as $K(x, y) = \Sigma_n \phi_n(x)\phi_n(y)$ which is computed to any structure $x, y \in X$.

2.2.1 Convolution Kernel

In Haussler [1999] following definition for *Convolution Kernel* shown:

Let $x \in X$ is a composite structure such that we can define $x_1 \ldots x_D$ as its parts(these parts can either be separate or overlapped parts). Each part is such that $x_d \in X_d$ for $d = 1, \ldots, D$ with $D \geq 1$ where each X_d is a countable set. Let R be the relation defined on the set $X_1 \times \ldots \times X_D \times X$ such that $R(x_1 \ldots, x_D, x)$ is true iff $x_1 \ldots, x_D$, are the parts of x. We denote with $R^{-1}(x)$ the inverse relation that yields that parts of x, that is $R^{-1}(x) = \{x_1 \ldots, x_D : R_(x_1, \ldots, x_D)\}$.

Then if there is a kernel K_d over $X_d \times X_d$ for each $d = 1, \ldots D$, and if two instances $x, y \in X$ can be decomposed in $x_1 \ldots x_d$ and $y_1, \ldots y_d$ then the following generalized convolution:

$$K(x,y) = \sum_{\substack{x_1, \ldots, x_d \in R^{-1}(x) \\ y_1, \ldots, y_d \in R^{-1}(y)}} \prod_{d=1}^{D} K_d(x_d, y_d)$$

is a valid kernel called a *convolution* or *decomposition kernel.* It is the zero-extension of K to $X \times X$ since $R^{-1}(x)$ is not not guaranteed to yield a non empty set for all $x \in X$.

In general this can be said as decomposition kernel is sum (over all possible ways to decompose a structured instance) of the product of valid kernels over the parts of instance.

Kernels can be customized for the requirement and in this experiment about *ncRNA secondary structure*s *kernel* choosen by the motivational paper [Costa and De Grave, 2010] suggested by [Haussler, 1999]

Definition 2.2.1. ***Kernel**: Let X be a set and $K : X \times X \rightarrow \mathcal{R}$, where $\mathcal{R}$ denotes the real numbers and $\times$ denotes set product. We say K is a kernel on $X \times X$ if K is symmetric, i.e for any x and $y \in X$, $K(x,y) = K(y,x)$, and K is positive definite, in the sense that for any $N \geq 1$ and any $x_1, \ldots, x_N \in X$, the matrix K defined by $K_{ij} = K(x_i, x_j)$ is positive definite, i.e. $\Sigma_{ij}\ c_i c_j K_{ij} \geq 0$ for all $c_1, \ldots c_N \in \mathcal{R}$.*

The definition can also be said, that if each $x \in X$ can be represented as $\phi(x) = \{\phi_n(x)\}_{n \geq 1}$ such that K is the ordinary l_2 dot product $K(x,y) = \langle \phi(x), \phi(y) = \sum_n \phi_n(x)\phi_n(y) \rangle$ then K is *kernel.*

Definition 2.2.2. ***Feature space**: If for a given kernel K can be represented as $K(x,y) = \langle \phi(x), \phi(y))$ for any choice of ϕ then X and given K are kernel. In specifically this is valid when for any kernel K over $X \times X$ where X is countable set. The vector space induced by ϕ called feature space.*

Feature space definition is comes after positive-semi definite that the *zero-extension* of a *kernel*: If $S \subseteq X$ is K is a kernel is a valid kernel, that is, if $S \subset X$ and K is a kernel on $S \times X$ by defining $K(x,y) = 0$ if x or y is not is S. It is easy to show that kernels are closed under summation i.e. a sum of kernel is a valid *kernel.*

2.3 NSPDK working

As we have seen the definitions and notations of graphs and kernel taken from Costa and De Grave [2010] here we define and discuss Neighborhood Subgraph Pairwise Distance Kernel (*NSPDK*). *NSPDK* is used for the work in the thesis. This is an *convolution kernel.*

Taking an instance of the decomposed *kernel* as follows:

Definition 2.3.1. ***NSPDK decomposed kernel****: It is defined as the relation $R_{r,d}(A^v, B^u, G)$ between two rooted graphs as seen in A^v and B^u and a graph G to be true iff both A^v and B^u ar ein $\{\mathcal{N}_r^v : v \in V(G)\}$, where we require that $A^v(B^u)$ be isomorphic to some $\mathcal{N}_r$ to verify the set inclusion, and that $\mathcal{D}(u, v) = d$. The relation $R_{r,d}$ selects all pairs of neighborhood graphs of radius r whose roots are at distance d in a given graph G*

It is formalized as that $k_{r,d}$ over $\mathcal{G} \times \mathcal{G}$ as the *decomposition kernel* on the relation $R_{r,d}$, and written as:

$$k_{r,d}(G, G') = \sum_{\substack{A_v, B_u \in R_{r,d}^{-1}(G) \\ A'_{v'}, B'_{u'} \in R_{r,d}^{-1}(G')}} \delta(A_v, A'_{v'})\delta(B_u, B'_{u'})$$

If $\delta(x, y)$ is the *exact matching kernel* then

$$\delta(x, y) = \begin{cases} 1 & x \simeq y \text{ (if the graph } x \text{ is isomorphic to } y\text{)} \\ 0 & \text{otherwise} \end{cases}$$

The above expression can be better understood if we look at the Figure 11 before proceeding to the computation steps where we require the collection of subgraph they are collected with above formulation. In the figure small circle which covers only one bond is of radius 1 and with distance d similarly the other 2 circles of radius 2 and 3. Here these radius is allowed to overlap on each other. *NSPDK*

With the parameters radius, distance, Graphs and kernel the Neighborhood Subgraph Pairwise Distance Kernel is defined as:

$$K(G, G') = \sum_r \sum_d k_{r,d}(G, G')$$

to increase the efficiency NSPDK uses zero-extension of K which is derived by limiting upper bound and the distance parameter with a limit the equation obtained as :

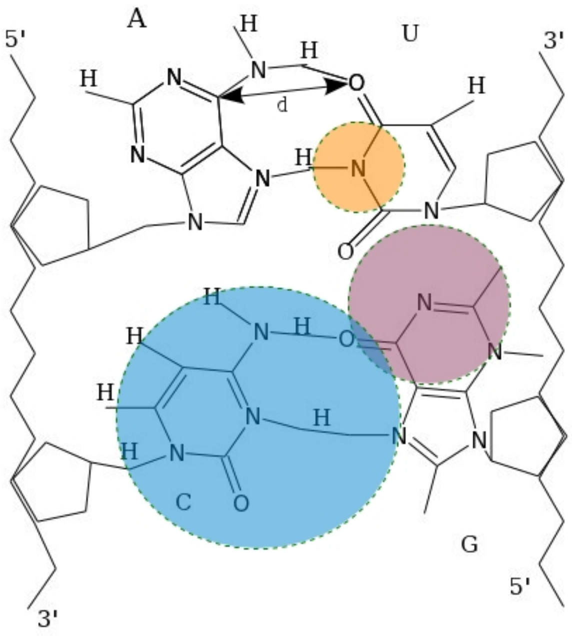


Figure 11: Neighborhood pairs in an typical AU-CG chemical structure showing with various radius orange(small circle), brownish(medium circle) and blue(large circle) each of 1, 2 and 3 radii respectively. d is the distance. Parameters constituting neighborhood graph

$K_{r^*,d^*}(G,G') = \sum_{r=0}^{r^*} \sum_{d=0}^{d^*} k_{r,d}(G,G')$

whereby NSPDK is limited to the sum of the $K_{r,d}$ kernels for all increasing values of the radius and the distance. parameter up to maximum given value $r^*(d^*)$. And a normalized version of a $K_{r,d}$ is taken to ensure that relations of all orders are equally weighted regardless of the size of the induced part sets. The normalized version of kernel is as follows:

$\hat{k}_{r,d}(G,G') = \frac{k_{r,d}(G,G')}{\sqrt{k_{r,d}(G,G)k_{r,d}(G',G')}}$

The secondary structures are interpreted as graphs interms of vertices with nucleotides and the edges forming the base pairs where if they exists.

To show the considered kernel is valid one it can be said that:

- The kernel is built as a decomposition kernel over the countable space over the countable space of all pair of neighborhood subgraphs of finite size; Which makes it to work all the neighborhood graphs and work on the radius taken.
- The zero-extension to bound values for the radius and distance parameters preserves the kernel property; This can be proved with the help [Haussler, 1999] as if $S \subset X$ and K is kernel on $S \times S$, then K may be extended to a kernel on $X \times X$ by defining $K(x,y) = 0$ if either x or y is not in S. This follows directly from the definition of a positive definite function. Then it is called zero-extension of K

NSPDK implements exact matching kernel $\delta(G_h, G'_h)$ in two steps.

1. A fast graph invariant encoding for G_h and $G'_{h'}$ via a label function $\mathcal{L}^g : \mathcal{G}_h \rightarrow$

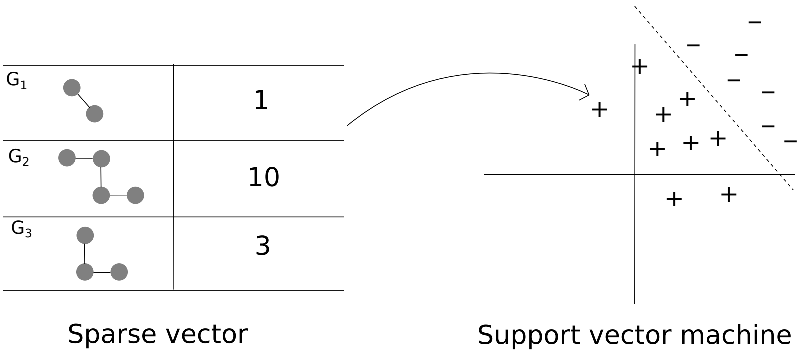


Figure 12: Graph index from sparse vector to support vector machine

$\sum^*$ where $\mathcal{G}_h$ is the set of rooted graphs and $\sum^*$ is the set of strings over a finite alphabet $\sum$;

2. It makes use of hash function $H : \sum^* \to \mathbb{N}$ to confront $H(\mathcal{L}^g(G_h))$ and $H(\mathcal{L}^g(G'_{h'}))$

With matching kernel an efficient string encoding of graphs from which a unique identifier via a hashing function from string to natural number. In this way the isomorphism test between two graphs is reduced to a fast numerical identity test, which is computed fastly with the better encoding string graph.

2.3.1 Graph Invariant and complexity

The graph encoding $\mathcal{L}^g(G_h)$ that is proposed described by introducing new label functions for vertices and edges, denoted $\mathcal{L}^n$ and $\mathcal{L}^e$ respectively. $\mathcal{L}^n(v)$ assigns to vertex v the concatenation of the lexicographically sorted listed of distance-label pairs $\langle \mathcal{D}(v,u), \mathcal{L}(u) \rangle$ for all $u \in G_h$.

With G_h being a rooted graph there will be knowledge about the identity of the root vertex h and include, for each vertex v, the additional information of the distance from the root node $\mathcal{D}(v,h)$.$\mathcal{L}^e(uv)$ assigns to edge uv the label $< \mathcal{L}^n(u), \mathcal{L}^n(v), \mathcal{L}(uv) >$.$\mathcal{L}^g(G_h)$ assigns to the rooted graph G_h the concatenation of the lexicographically sorted list of $\mathcal{L}^e(uv)$ for all $uv \in E(G_h)$. In words: we relabel each vertex with a string that encodes the vertex distance from all other labeled vertices (plus the distance from the root vertex); the graph encoding is obtained as the sorted edge list, where each edge is annotated with the endpoints' new labels.

Time complexity is dependent on procedures:

- the extraction of all pairs of neigh borhood graphs $\mathcal{N}^v{}_r$ at distance $d = 0, \ldots, d^*$,
- computation of the graph invariant for those subgraphs.

The first procedure is addressed by implementing factoring it into the extraction of $N^v{}_r$ for all $v \in V(G)$ and the computation of distances between pairs of vertices whose pairwise distance is less than d^* . For next step breadthfirst (BF) is repeated.

For the complexity issue can be analyzed in terms of one the computation of the string encoding $\mathcal{L}^g(G_h)$ and another with the computation of the hash function $H(L^9(G_h))$.

In the first part it is dominated by the computation of all pairwise distances in O(|V (G_h)||E(G_h)|) and the sorting of the relabeled edges, which has complexity O(|V (G_h)||E(G_h)| log |E(G_h)|) since edges are relabeled with strings containing the distance information of the endpoints from all other vertices. The hash function complexity second and the last part is linear in the size of the string.The overall complexity O(|V (G)||V (G_h)||E(G_h)| log |E(G_h)|) is dominated by the repeated computation of the graph invariant for each vertex of the graph. Since this is a constant time procedure for small values of d^* and r^*, it is concluded that the NSPDK complexity is in practice linear in the size of the graph.

To reduce space complexity, the hash collisions are monitored, as this would force the algorithm to keep in memory all the encoding key hashed value pairs.

For improving the NSPDK the another version of NSPDK is SVMSGDNSPDK (Support Vector Machine Stochastic Gradiant Descent Neighborhood Subgraph Pairwise Distance Kernel) more on this is seen in section 3.5 used which utilized in this thesis for the computation purposes of the accuracy measures of *proposed structure* obtained from *RNAshapes* and the *true structures* obtained from *Rfam* the program takes the help of sparse vector and support vector machine which is as shown in Figure 12.

Initially all the rooted graphs are generated and are given are arranged in an sparse vector with their weightage these are then mapped with support vector machine.

3 Experiments

3.1 Data sources

In this thesis we can see, various *RNA* data from various data sources couple of them are taken and are briefed.

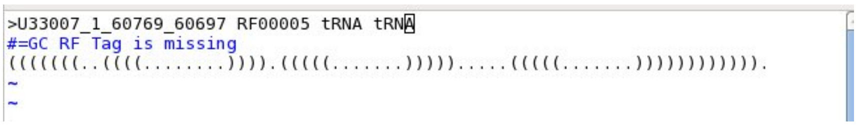
3.1.1 Rfam

It is a comprehensive collection of *non-coding RNA* (*ncRNA*) families, which are represented by multiple sequence alignments and their profile are with stochastic context-free grammar SCFG. More details are explained in the reference [Griffiths-Jones et al., 2005]. And the specification of the database in Stockholm format [Wiipedia, 2012b] explained in detail in the reference [Gardner et al., 2011].

Database	Reference	URL
RNA Strand	[Andronescu et al., 2008]	http://www.rnasoft.ca/strand/
Rfam	[Griffiths-Jones et al., 2005] [Gardner et al., 2011]	http://rfam.sanger.ac.uk/

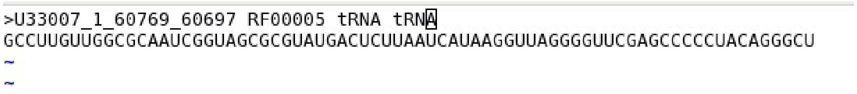
The stockholm format data is taken according to from the Rfam seed 10.1 version which has 1973 accession key numbers. The data for the purpose of the experiment is taken in the Fasta format (subsection 3.1.1) so that it can be used with the tool *RNAshapes*. Same data is also taken in the form of DBN format (subsection 3.1.1) so that base pairs can be processed for computations.

textbfDot-Bracket Notation (DBN): It is a Text based format file. To represent *RNA secondary structure* in recently it is represented in newly developed format called *Dot-Bracket Notation* (DBN) or it is also called as Dot-Parentheses (DP) format. These are well parenthesized words and has dots *'.'*, opening bracket *'('* and closing bracket *')'*. Dotted positions are unpaired nucleotides and brackets takes part forming a pair of nucleotides with each other which is called as *basepair*. As the *basepairs* are formed in pairs it is expected to have brackets in pairs. In DBN *Pseudoknots* are marked using alternative [*ldots*] or with {*ldots*} bracket pairs. As this is widely used in *RNA structure* and so nucleotides (*A, C, G, U* are represented as DBN). These kind of DBN are widely used to build the *RNA secondary structure* and also used to plot *RNA secondary structure* plots with tools such as *RNAplot.*



```
>U33007_1_60769_60697 RF00005 tRNA tRNA
#=GC RF Tag is missing
(((((((..((((........)))).(((((.......))))).....(((((.......)))))))))))).
~
~
```

Figure 13: In the first line usually written some information related to the sequence.



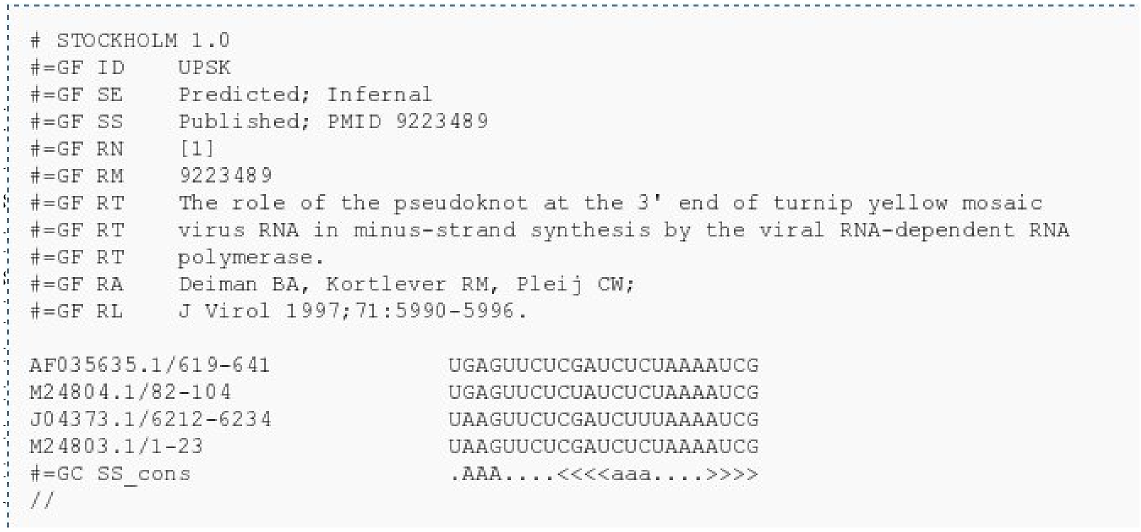
```
>U33007_1_60769_60697 RF00005 tRNA tRNA
GCCUUGUUGGCGCAAUCGGUAGCGCGUAUGACUCUUAAUCAUAAGGUUAGGGGUUCGAGCCCCCUACAGGGCU
~
~
```

Figure 14: The first line starts with '>' symbol. The first word after the symbol is the name of the sequence. and after it is customized information.

A typical DBN file can be seen in Figure 13. DBN format does not yet have a standardized sequence.

Fasta format: Sequence of fasta format starts with single line description, followed by lines of sequence data. To differentiate first line of fasta format it starts with greater-than symbol (“>”). Every sequence requires an identifier which is specified right after greater-than symbol. First word after the symbol is the sequence identifier. The line also consists other information such as External source which specifies about the data bank, number of molecules.

Stockholm format: Sequences which are obtained from Rfam database are of stockholm format and are multiple sequence alignment format. these format are more in use for RNA sequence alignments. The files of Rfam are generated by Infernal tool. A sample format can be seen in Figure 15



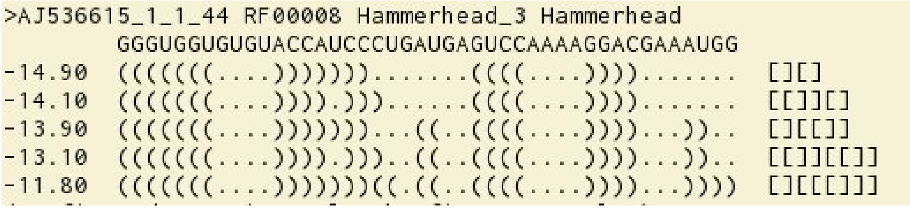
```
# STOCKHOLM 1.0
#=GF ID    UPSK
#=GF SE    Predicted; Infernal
#=GF SS    Published; PMID 9223489
#=GF RN    [1]
#=GF RM    9223489
#=GF RT    The role of the pseudoknot at the 3' end of turnip yellow mosaic
#=GF RT    virus RNA in minus-strand synthesis by the viral RNA-dependent RNA
#=GF RT    polymerase.
#=GF RA    Deiman BA, Kortlever RM, Pleij CW;
#=GF RL    J Virol 1997;71:5990-5996.

AF035635.1/619-641             UGAGUUCUCGAUCUCUAAAAUCG
M24804.1/82-104                UGAGUUCUCUAUCUCUAAAAUCG
J04373.1/6212-6234             UAAGUUCUCGAUCUUUAAAAUCG
M24803.1/1-23                  UAAGUUCUCGAUCUCUAAAAUCG
#=GC SS_cons                   .AAA....<<<<aaa....>>>>
//
```

3.2 Proposed Structure models

In *RNA* field, there are many structure prediction software which work with various strategies [Wiipedia, 2012a] a couple of them are considered in this thesis. As *RNAshapes*

3.2.1 RNAshapes

RNAshapes is mainly dependent on MFE. As seen in subsection 1.3.1 with the evolution of various methods RNAshapes also took advantage of these methods and formulated *secondary structure* prediction tool. It also considered some of the available prediction tools to look into its drawbacks and to improve them. Considering one of the suboptimal method in which an algorithm was developed [WUCHTY et al., 1999] allowed structures which are not redundant and they have complete suboptimal folding is implemented in the tool *RNAsubopt* which is part of the *Vienna RNA* package [Hofacker et al., 1994], which is mainly designed not to miss any structure which is plausible with respect to the nearest neighbor energy model. It considers all the structures which are in the given range of energy. and helps in viewing all possible *subptimal* structures whereby the percentage of predicted structures similarity to the secondary structures increases. However, the computation problem may arise with the exponential increase of suboptimal foldings as the length increases [Smith and Waterman, 1981] and produces large number of structures.



```
>AJ536615_1_1_44 RF00008 Hammerhead_3 Hammerhead
        GGGUGGUGUGUACCAUCCCUGAUGAGUCCAAAAGGACGAAAUGG
-14.90  (((((((....))))))).......((((....)))).......  [][]
-14.10  (((((((....)))).)))......((((....)))).......  [[]][]
-13.90  (((((((....)))))))...((..((((....))))...))..  [][[]]
-13.10  (((((((....)))).)))..((..((((....))))...))..  [[]][[]]
-11.80  (((((((....)))))))((.((..((((....))))...))))  [][[[]]]
```

Figure 16: RNAshapes output with abstract structure at level 3

endcenter

RNAshapes has a concept called *abstract shapes* which are generic concept. As described in [Voß et al., 2006] they are defined by the means of abstraction functions preserving varying amount of detail. These functions are homomorphisms from structure to another tree-like domain, along with preserving the adjacency and nesting of substructures for the trees representing shapes four operators are used such as OP (“open”) which represents the shape of all structures without base pairs, CL (“closed”) represents helocal region and AD and E are re-used. And to represent the lists of adjacent (sub)shapes, *RNAshapes* two abstraction functions such as $\phi 5$ and $\phi 3$. Where in this thesis experiment abstraction level $\phi 3$ is used.

The complete experiment process can be looked in the given flowchart Figure 18 where it can be seen that the data is taken from *Rfam* databank. Which is of

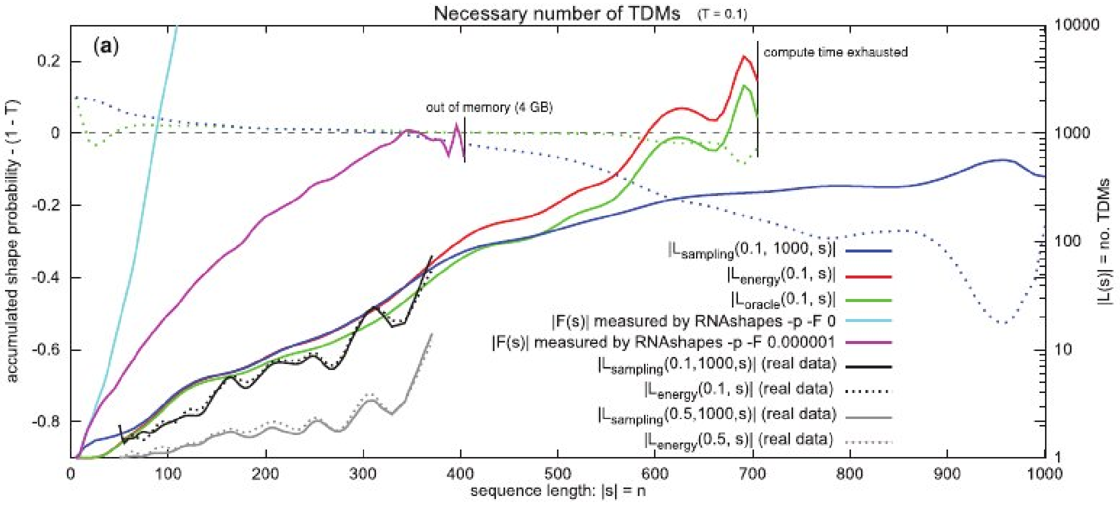


Figure 17: Graph showing Probability along with sequence length for the RNAshapes taken from [Janssen and Giegerich, 2010]

stockholm format seeds. The consensus structure from the stock holm format is converted into fasta format so that it can be given as input to RNAshapes in the next step with parameters which are choosen to be as follows.

- Abstract Type: This for the entire experiment choosen to be 3 which abstract level 3 of RNAshapes.
- Number of desired sequences to limit: It is taken as a standard of 20 although many sequences were not producing 20 various outputs.
- Energy range: This sets the energy range as percentage value of the minimum free energy. For example, when -c 10 is specified, and the minimum free energy is -10.0 kcal/mol, the energy range is set to -9.0 to -10.0 kcal/mol. In the experiment various range of energy tried.

A sample ouput from RNAshapes can be seen in Figure 16

3.3 Measures of Accuracy

In the experiment as we have taken The measures of accuracy which are considered in this thesis are as follows

Sensitivity: It is a statistical measure considered to measure the proportion of actual positives which are correctly identified. This is taken into consideration to get measures between *true structure* taken from *Rfam*(see subsection 3.1.1) and the *proposed structure* which is obtained by *RNA* secondary structure tool. As discussed in subsection 3.2.1.It is measured as

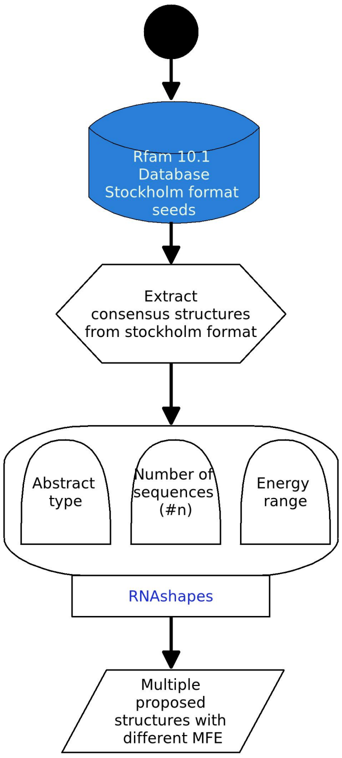


Figure 18: Flowchart showing Rfam database to *secondary structure* prediction with RNAshapes with some useful options.

$$Sensitivity = \frac{\text{number of correctly predicted base pairs}}{\text{number of true base pairs}}$$

Positive predictive value(PPV): It is also statistical measure which is positive predictive index value or the precison rate. It is the proportion of subjects with positive test results who are correclty diagnosed it is a criticial measure of the performance of a diagnostic method. As it reflects the probability that a positive test reflects the underlying condition being tested for its value however depen on the prevalence of the outcome of interest. It is measured as follows:

$$PPV = \frac{\text{number of correctly predicted base pairs}}{\text{number of predicted base pairs}}$$

F-Measure: It is defined as the haromonic mean of precision$P$ and the recall R[1]. F-Measure combines both the values of Sensitivity and the PPV. It can be measured as follow:

$$F\text{-}Measure = \frac{\text{number of True positives}}{\text{number of True positives + number of false Positives}}$$

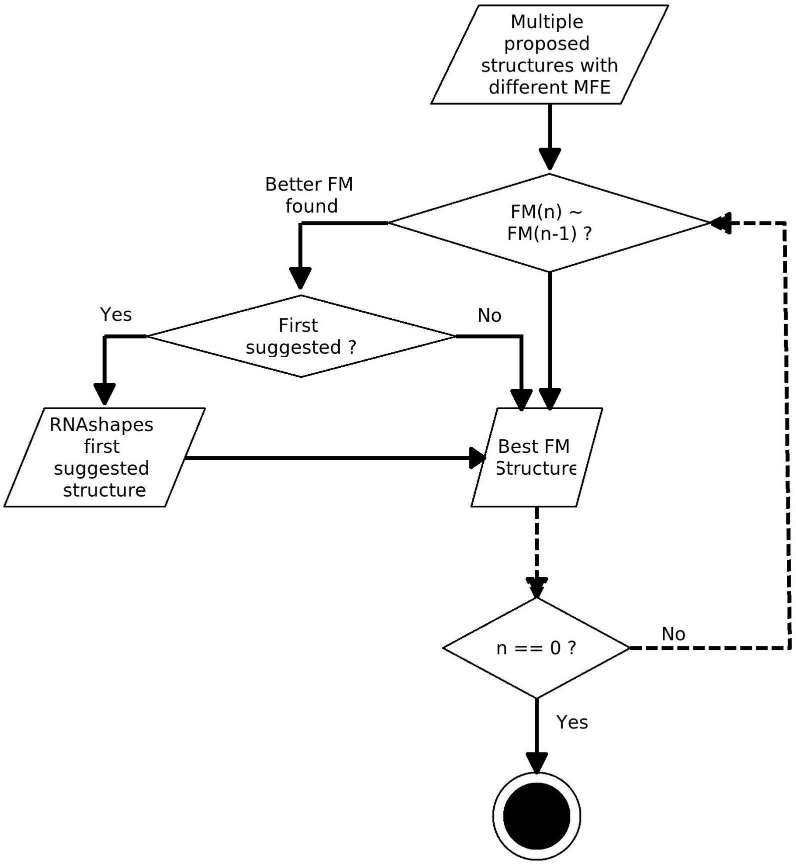


Figure 19: Flowchart: Accuracy evaluation between RNAshapes best suggested and the best structure available

3.3.1 Accuracy

In the experiment at various steps accuracy measure is calculated such as after generating a *proposed structure* from *RNAshapes* it is compared with true structure for the number of pairs obtained when compared to the *consensus structure.* However as the *RNAshapes* gives multiple predicted structures varying in their energy values. The flow chart Figure 19 shows how the this process is carried out where if a the best possible measure and the RNAshapes suggested are same then it they both have the same value other wise they vary in their values and this information is stored in the training and target files which is later utilized in the training and testing process.

Receiver operating characteristic: It is also simply specified as ROC: It is a graphical plot of the sensitivity, or true positive rate. vs false positive rate(one minues the specificity or true negative rate) for a binary classifier system as its discrimination threshold varied.

This measure mainly used to know how better is the training action on another object so that what kind of measures can be accepted and what kind of measures be discarded.

In the classifier model or the classifier diagnosis mapping of instances between certain classes is performed and here in the experiemnt it gives the information about an instance of learning is good enough to learn and to make model out of it.

3.4 Data Format

For the analysis of the data obtained from *RNA* Strand and Rfam databases (**??**), have been taken and are customized for processing and the formats followed are described in the following sections:

3.4.1 Stochastic gradient descent

It is a measure mostly utilised in machine learning and statisticians which gives an estimation of how to minimize objective function that has the form of a sum. Where the parameter w is to be estimated and where typically each summond function is associated with the i^{th} observation in the data set(used for training) as said in]

In classical statistics, sum minimization problems arise in least squares and in the maximum likelihood estimation for an independent observation. The general class of estimators that arise as minimizers.

This is used in the extended tool of NSPDK section 2.3. The usage of this tool is seen in the form of flowchart Figure 20 and can be seen in the following section.

3.5 Appliying Kernel Model

With the help of extended tool SVMSGDNSPDK which is Support vector machine Stochastic Gradiant Dessent the process.

In this the results which are obtained from the previous process where the accuracy is calculated and we obtain the best possible prediction of a sequence and also an information is taken from the previous process tool that what is the best possible value obtained from the tool all these values are given to SVMSGDNSPDK process.

As seen in the NSPDK it uses all those and uses support vector machine to map the sub graphs information. These information is fed in the form of graph data which is also called as gspan format Yan and Han [2002] also a target data is given which has the same information but in target format for the program.

gspan format which is in accordance with NSPDK is defined as follows:

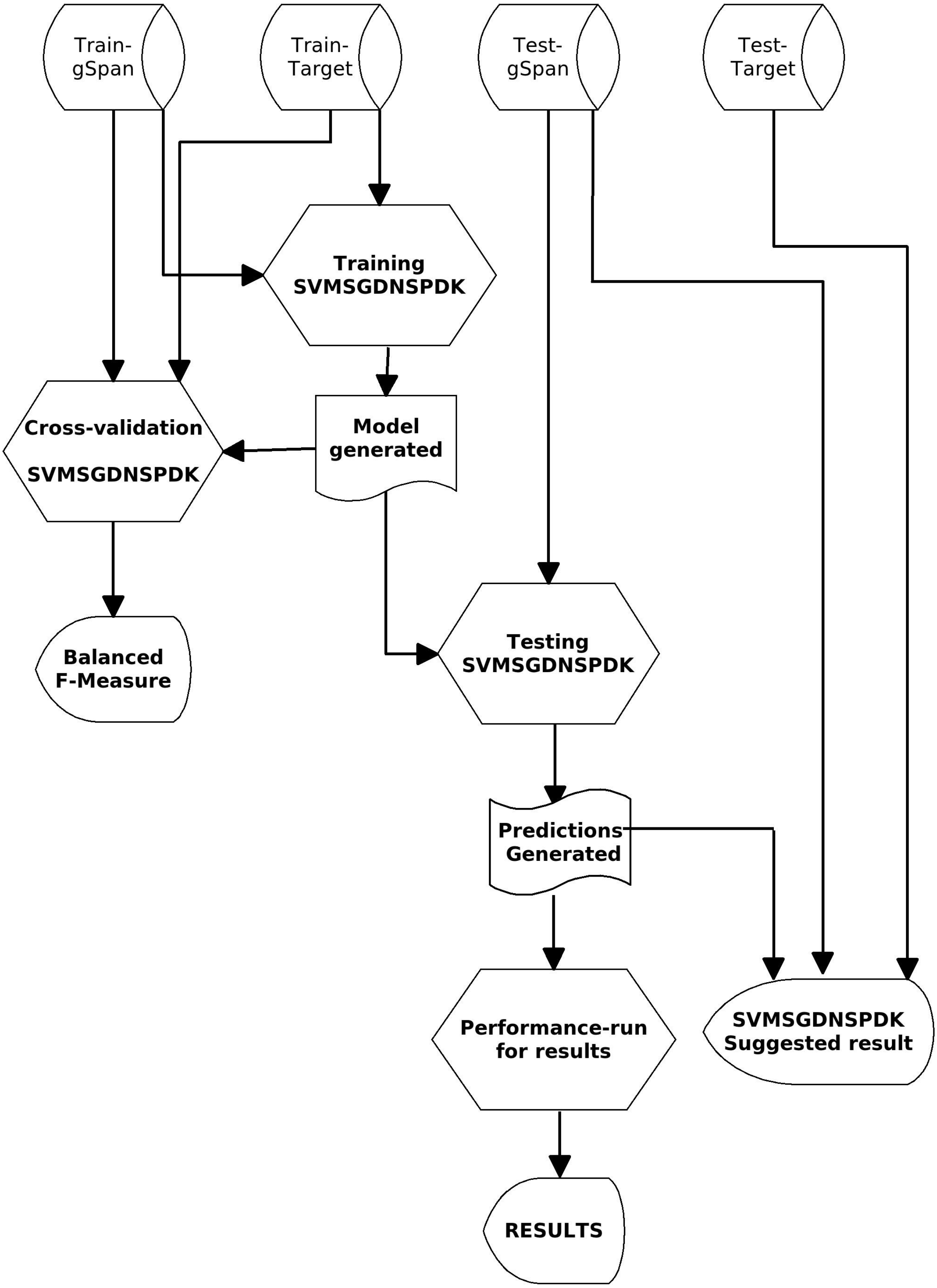


Figure 20: Flowchart showing how processing is done with SVMSGDNSPDK.

Definition 3.5.1. *Labeled Graph: A labeled graph can be represented by a 4-tuple,* $G = (V, E, L, L)$ *where*

V *is a set of vertices* $E \subseteq V \times V$ *is a set of edges,* L *is a set of labels,* $lV \cup E \rightarrow, l$ *is a function assigning labels to the vertices and the edges.*

All the graph functionalities which are taken for NSPDK are valid for SVMS-GDNSPDK

With this kind of setup it is possible to have better Support vector machine.which can be mapped from sparse vector. At fist the training data is given to the Training phase to get a model here it contain mainly which are positive responses are taken and also upon supervised learning all the favorable results are taken. Now this model is given to the testing phase along with all the unfavorable results so that the tool based on model can suggest its predictions. A file containing all the test values and prediction values is generated. Here now one can calculate performance measures with the performance such as ACC, PRF, APR and ROC. Where ROC as we have seen in the previous section gives us more information about the best useful data.

Now with predictions obtained and the test objects one can obtain the predicted or the suggested values by the program.

Results can be observed in the next secton.

3.6 Graphs and Results

Following are the results obtain in the form of graphs for the experiments performed and they are consolidated image of measures consisting of all three measures of best, learned and RNAshapes.

More graphs are kept in Appendix. Appendix A

3.7 Evaluation

On seeing the results it some of the following evaluations can be done:

- We see that SNORD Clan family produced better improvment in F-Measure.
- They were producing better results when the size and energy range were increasing.
- Although there are good improvement upto 375% but there are also some negative values in training for the same CLAN such as SNORD62_clan

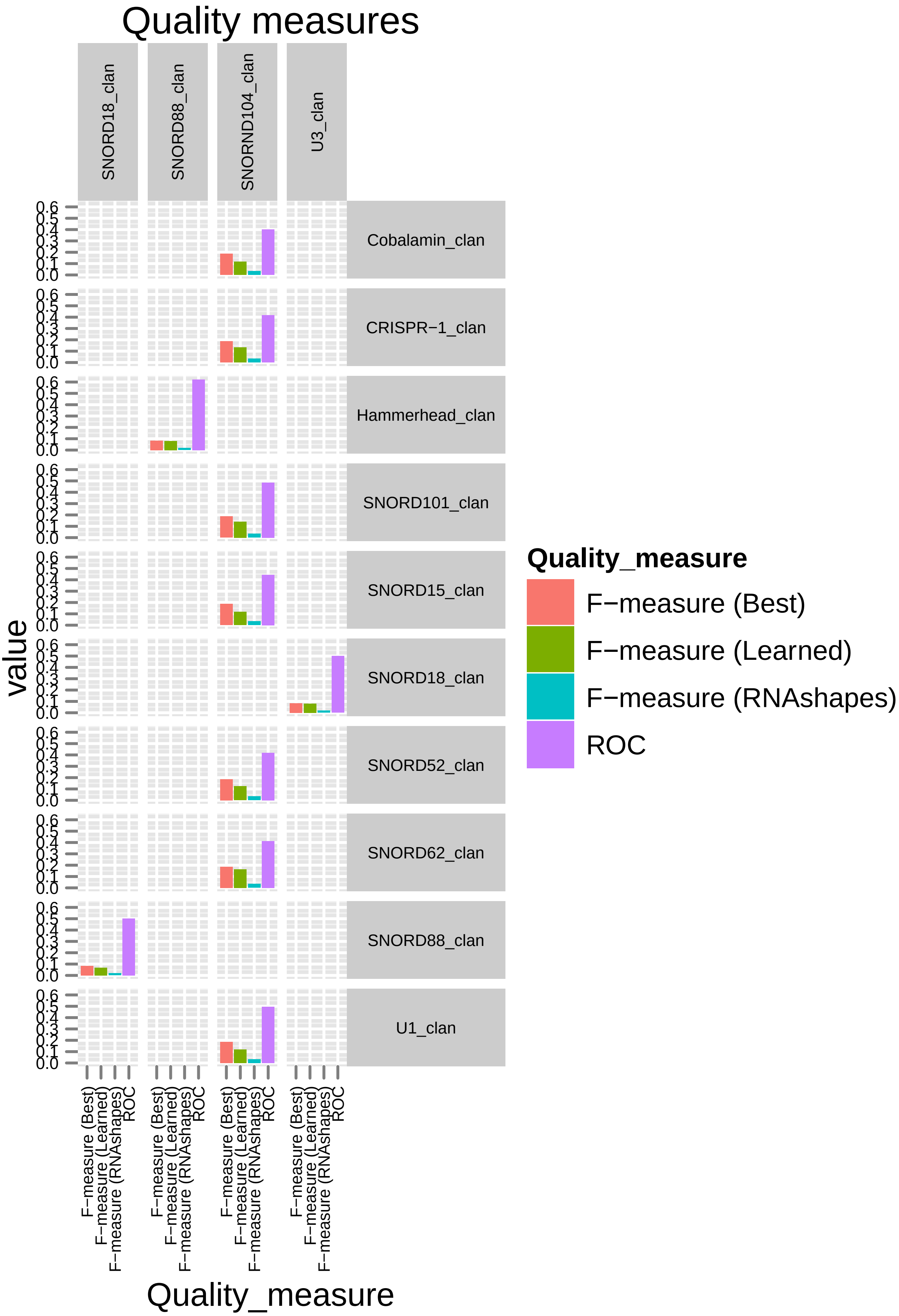


Figure 21: Graph shows major clans which have gained from the train and test process

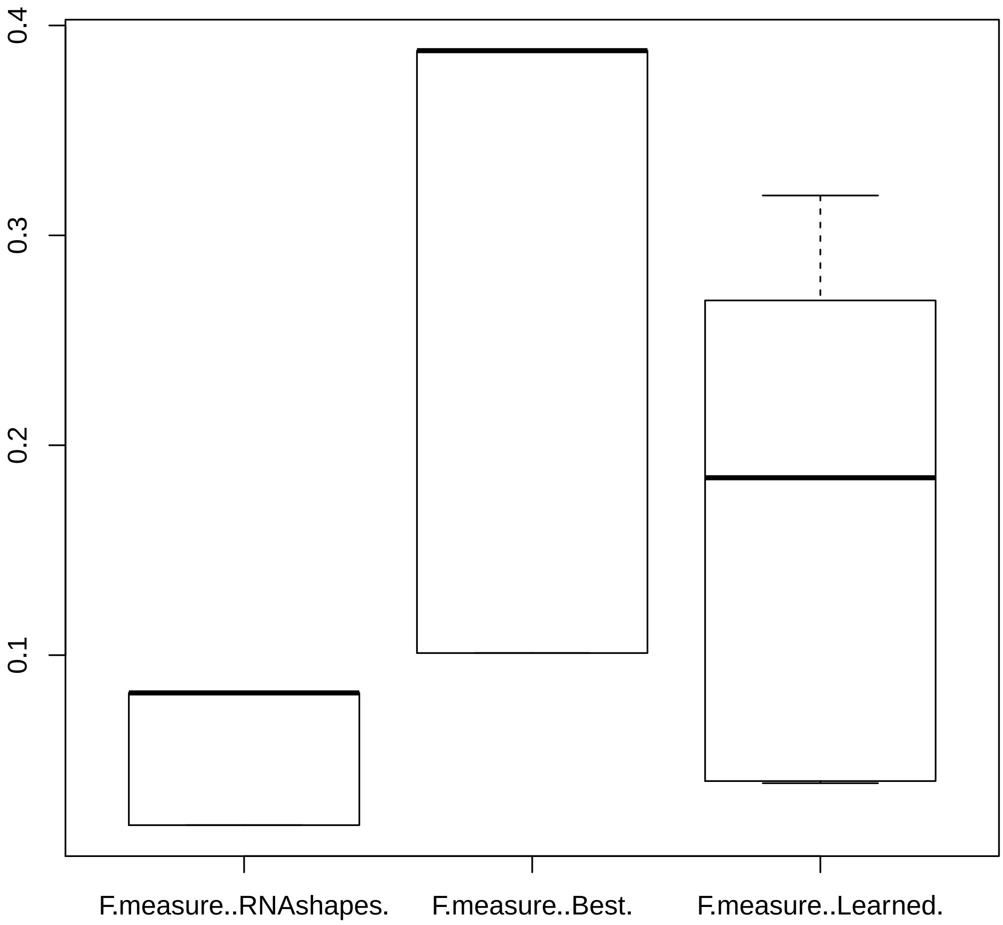


Figure 22: Graph shows major clans which have gained from the experiment SVNS-GDNSPDK

Train Clan	Test Clan	Nr of Nucleotides	Energy	F-measure (RNAshapes)	F-measure (Best)	F-measure (Learned)
SNORD62 clan	CRISPR-2 clan	100	40	0.825	0.91	0.878
CRISPR-1 clan	SNORD101 clan	100	40	0.082	0.388	0.329
CRISPR-1 clan	SNORD101 clan	100	80	0.082	0.388	0.319
SRP clan	SNORD101_clan	100	80	0.082	0.388	0.269
Hammerhead clan	SNORD88 clan	100	15	0.019	0.083	0.08
SNORD101 clan	tRNA clan	100	15	0.019	0.079	0.077
SNORD105 clan	SRP clan	100	10	0.019	0.079	0.065
Hammerhead clan	SNORD101 clan	300	5	0.079	0.152	0.123
SNORD110 clan	SNORD101 clan	100	5	0.079	0.152	0.119

4 Conclusion & Discussion

The problem of determining of a better and best *ncRNA secondary structure* is in increasing demand due to its various applications in the field of medicine and chemioinformatics, for therapeutics and diagnosis that target *RNA* problem in the area of bio-informatics which can be overcome to a certain extent with the introduction and penetration of *machine learning* tools which have successfully traversed in many other areas to increase their automated improvement.

After seeing the experiments in the thesis it can be seen that *proposed structured models* of various tools has much scope to be improved, which can be addressed machine learning tools *kernel* and *graph structures* procedures which uses technique of exact matching between pairs of small *isomorphic graphs* where the *secondary structure* problem of *RNA* similar structures can be solved to a fair extent. As the tool uses better equipped fast graph invariant procedures gives the scope of solving things better time complexity. This property is highly useful to the huge data such as *Rfam*(*ncRNA database*) full seed databases, taking inspiration from the results generated from their seeds.

With several runs and more better analysis one can recognize better *training models* which can serve as better models in *testing*

4.1 Futuristic View

Having seen with the experiment results that there is a very good scope with kernel models to improve the prediction structure accurac of the *ncRNA* sequences, in future one can use such tools. In particularly tools with graph kernels have much scope to customize and to provide better analyais in the learning process.

As there manay sequences and huge databases of ncRNA if one runs more extensively on things then there is every chance of making more educated analysis.

There are real good reasults with certain Clan familes such as SNORD18_clan but there also same time within same type of family they are very bad. Although reason are not clearly know further experiments on the full seed data rather than seeds might give more results in future.

Acknowledgments

Primarily I thank Dr. Fabrizio Costa for his guidance and supervision, he just not guided me in my work but also gave me great ease while doing my work and in problem solving situations. More importantly his works gave motivation to my thesis work.

I am very glad to get a topic in the Bioinformatics Chair under reverent and inspirational Prof. Dr. Rolf Backofen guidance under whom I have taken couple of BioInformatic courses.

I Thank IndiaYouth.info group which gave me inspiration in pursuing my masters and also gave motivation and platform on how to further take up my knowledge and enlightenment to the people in future.

Finally and not the least my dear parents and my dearest sister Jyothsna Manjunath Gupta.

A Appendix

More experiment results. in graphs

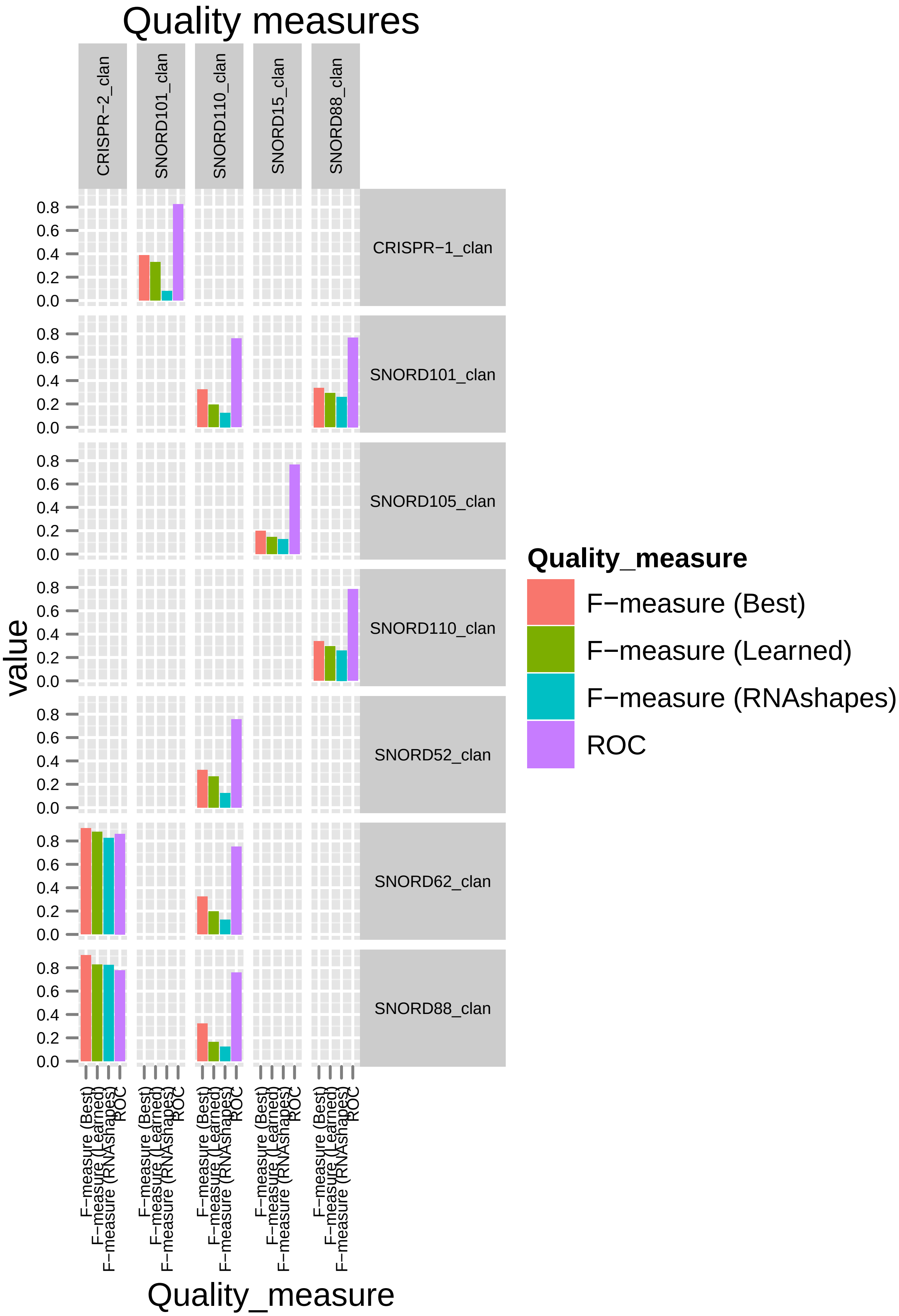


Figure 23: Graph shows major clans which have gained from the train and test process

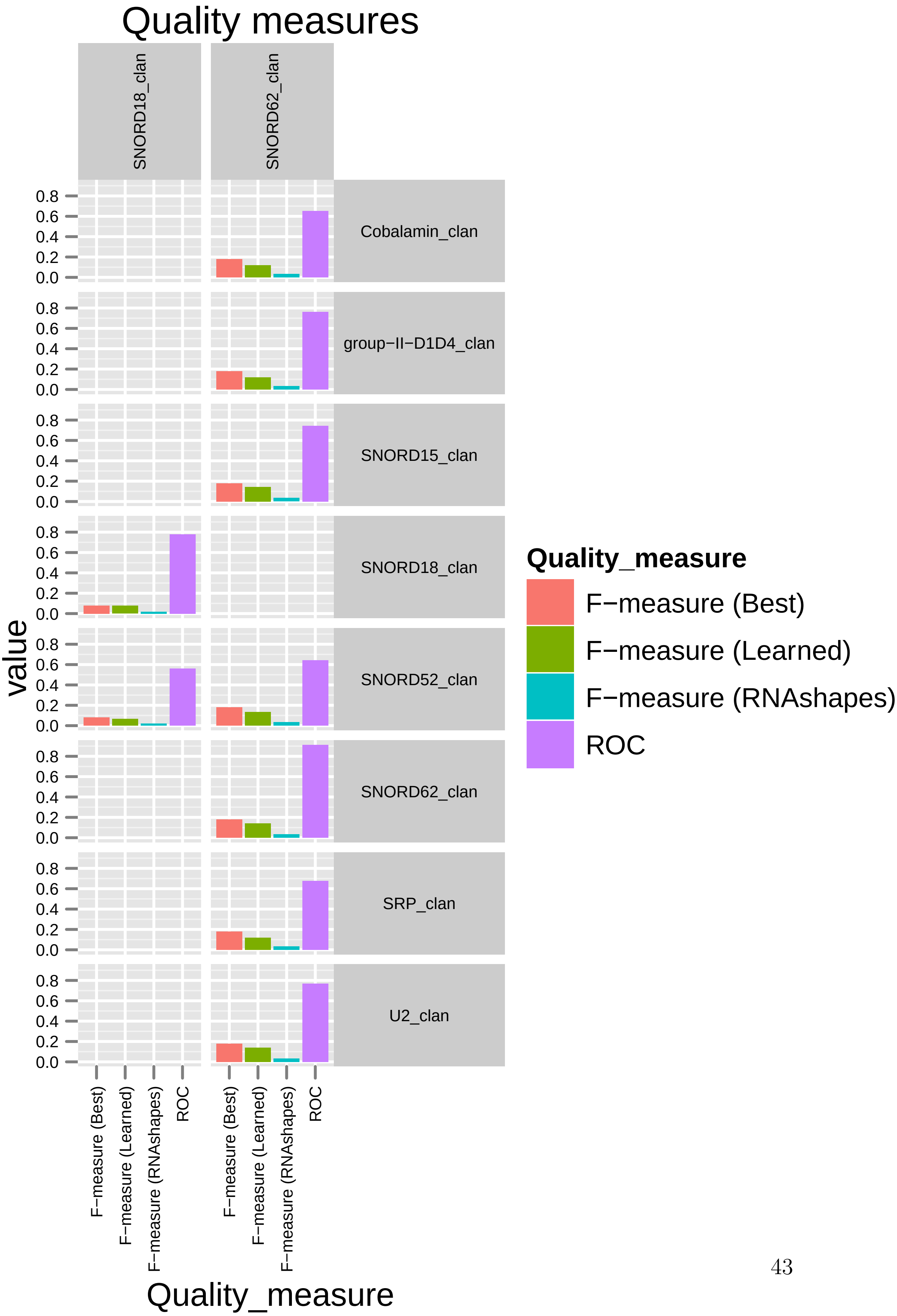


Figure 24: Graph shows major clans which have gained from the experiment SVNS-GDNSPDK

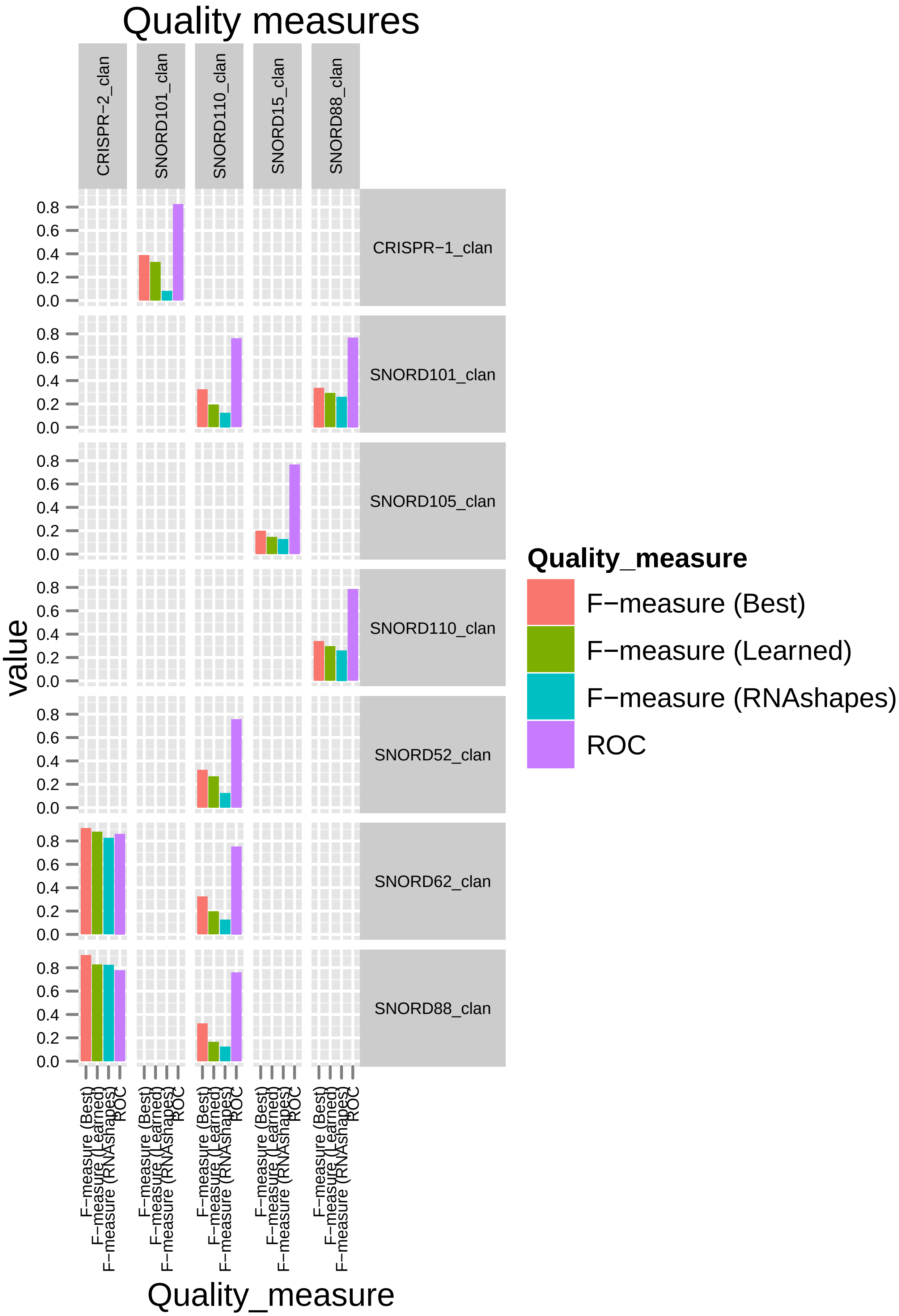


Figure 25: Graph shows major clans which have gained from the train and test process

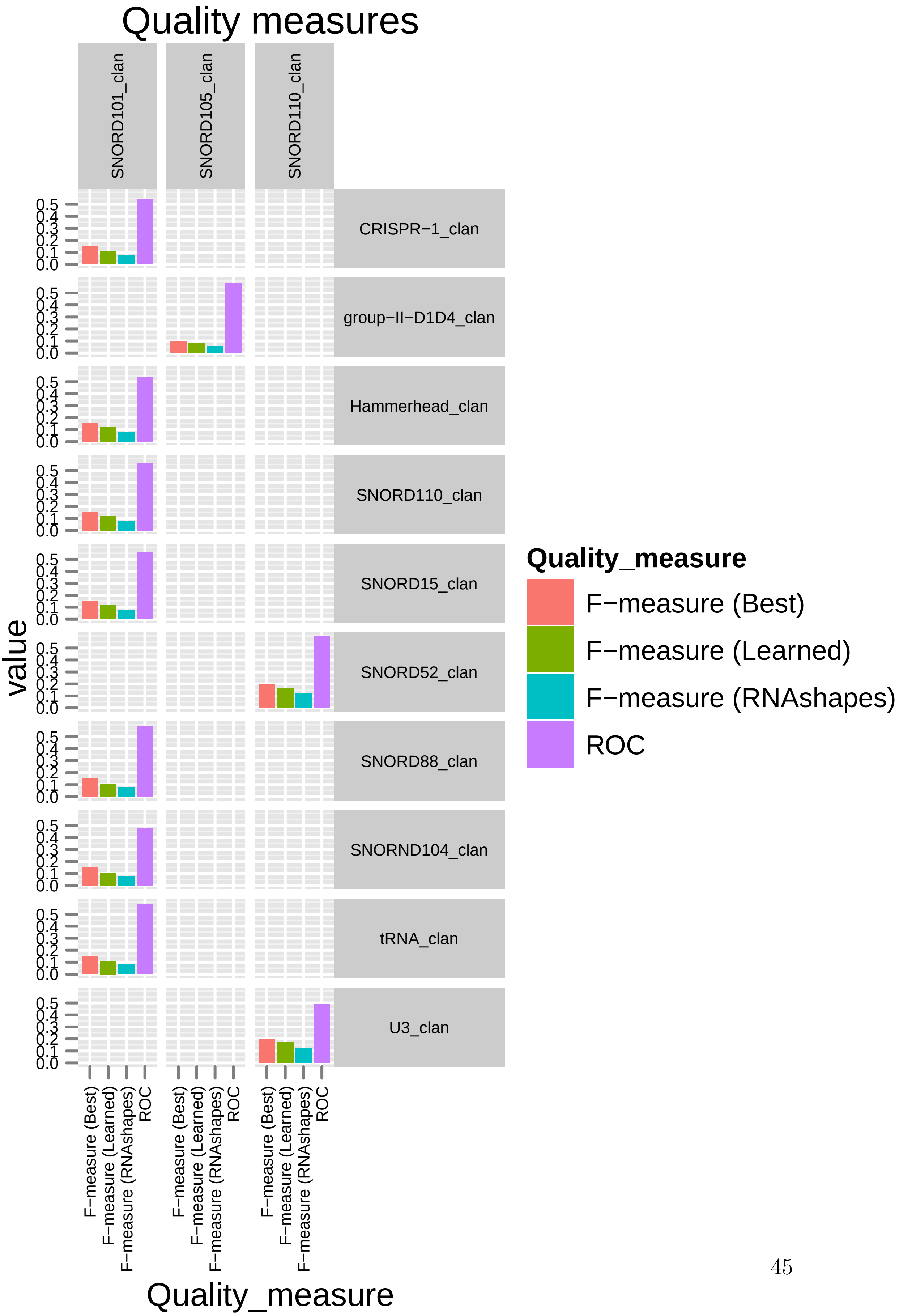


Figure 26: Graph shows major clans which have gained from the experiment SVNS-GDNSPDK

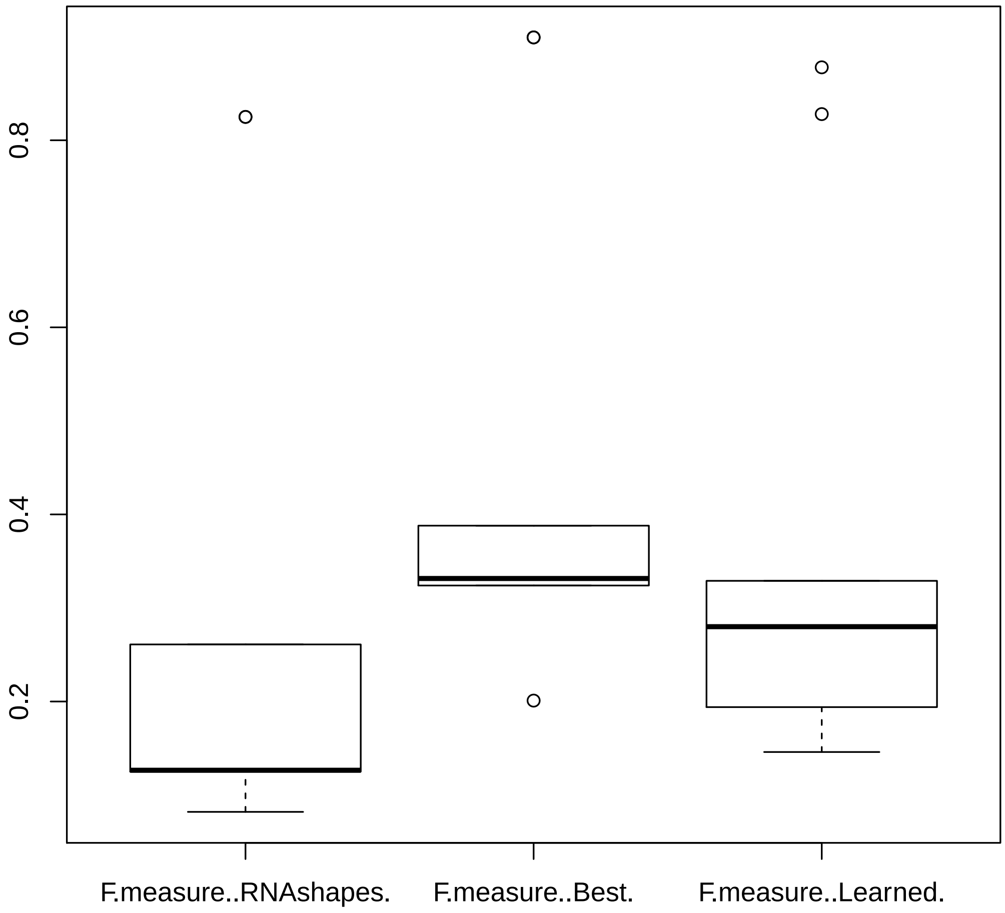


Figure 27: Graph shows major clans which have gained from the train and test process

Histogram showing various measures at 200 maximum length of nucleotides and with 15 energy range given to RNAshapes to produce the output

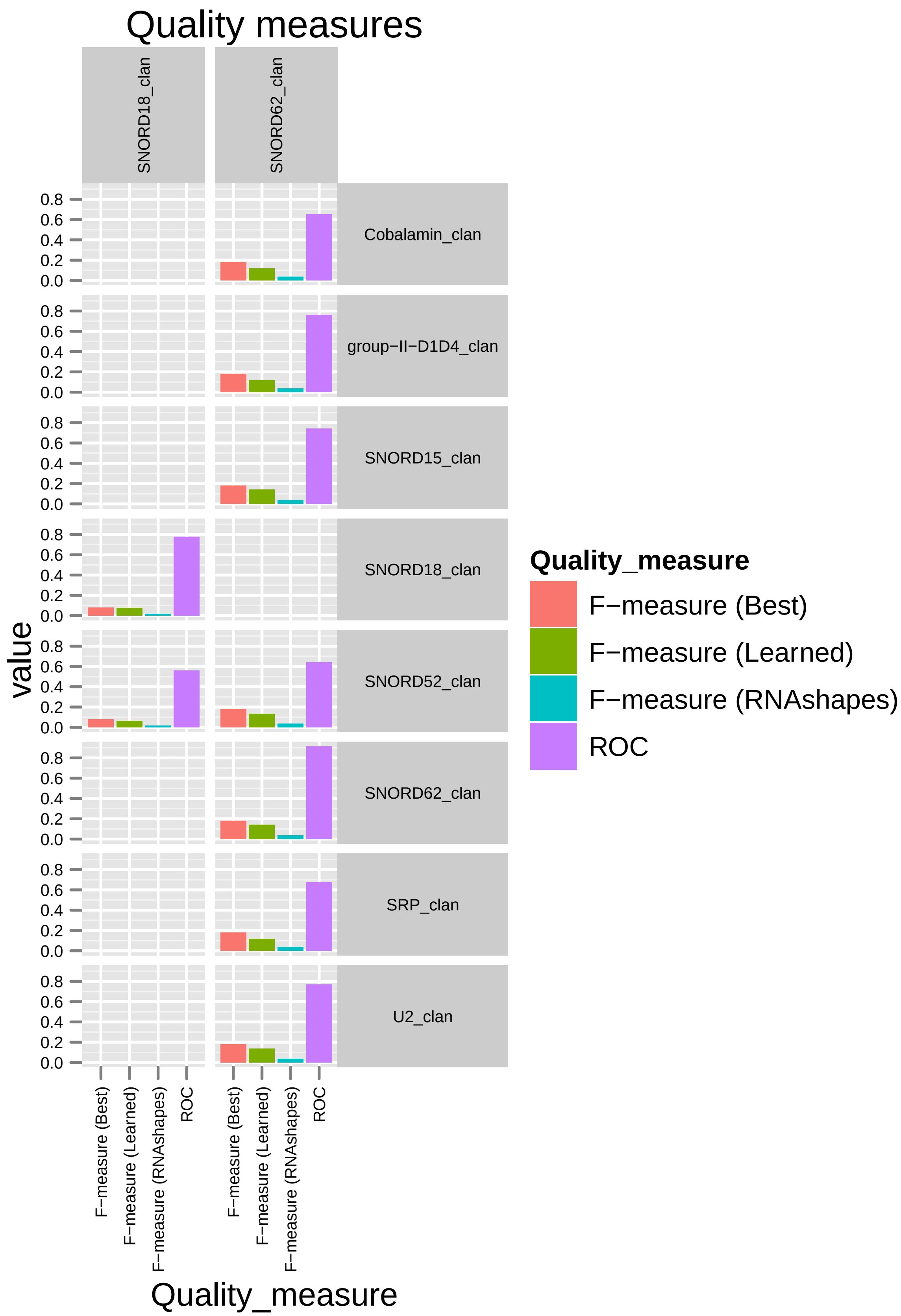


Figure 28: Graph shows major clans which have gained from the experiment SVNS-GDNSPDK

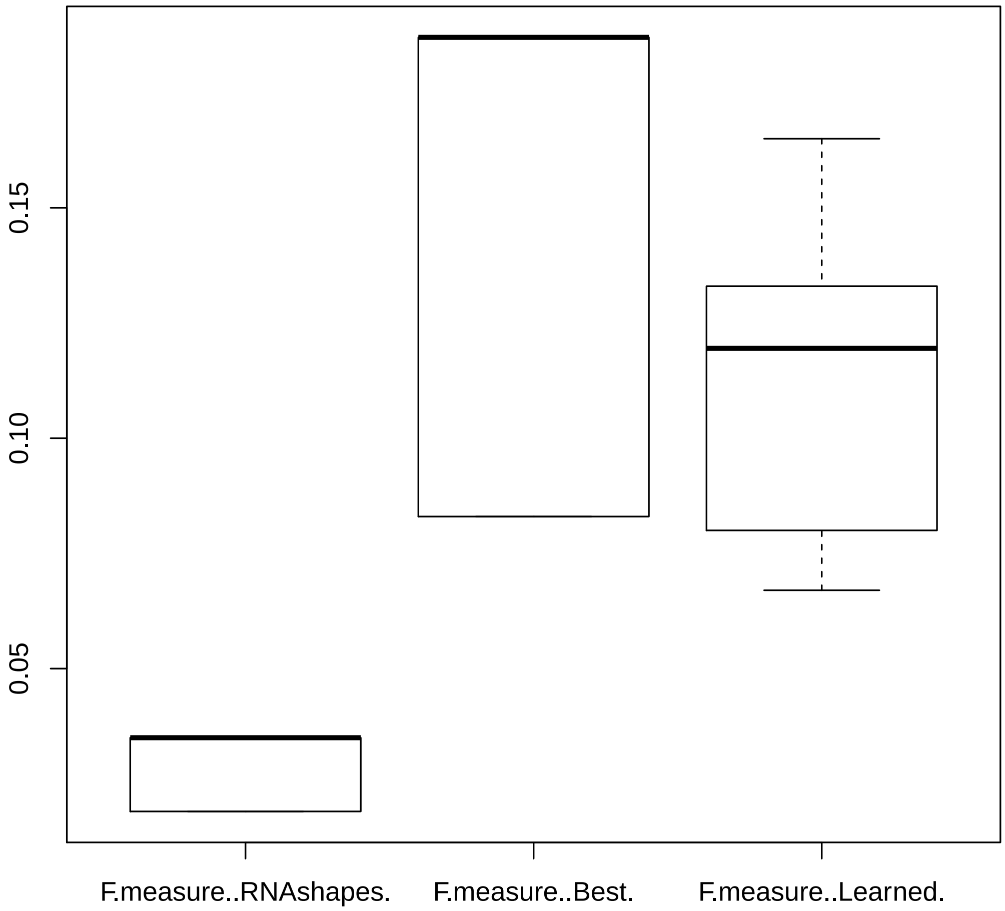


Figure 29: Graph shows major clans which have gained from the train and test process

Histogram showing various measures at 200 maximum length of nucleotides and with 15 energy range given to RNAshapes to produce the output

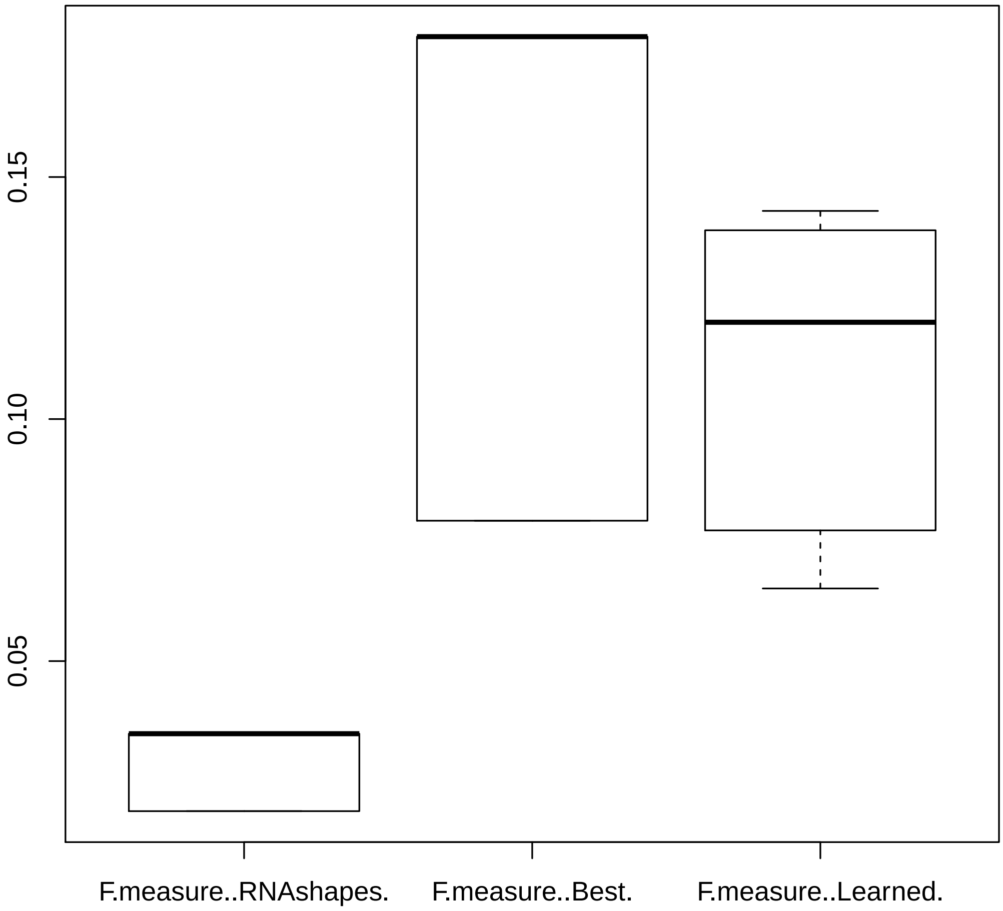


Figure 30: Graph shows major clans which have gained from the experiment SVNS-GDNSPDK

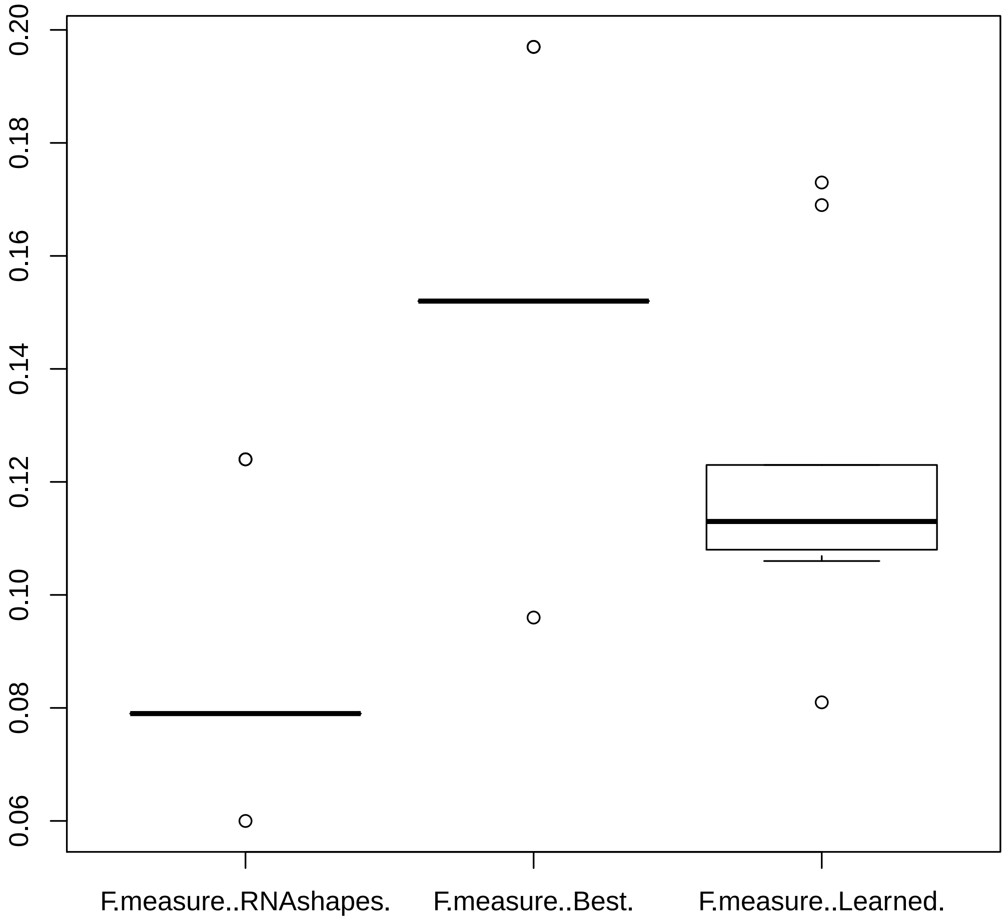


Figure 31: Graph shows major clans which have gained from the train and test process

Histogram showing various measures at 200 maximum length of nucleotides and with 15 energy range given to RNAshapes to produce the output

Bibliography

M. Andronescu, V. Bereg, H. Hoos, and A. Condon. Rna strand: the rna secondary structure and statistical analysis database. *BMC bioinformatics*, 9(1): 340, 2008. doi: 10.1186/1471-2105-9-340. URL `http://dx.doi.org/10.1186/1471-2105-9-340`.

T.H. Byers and M.S. Waterman. Determining all optimal and near-optimal solutions when solving shortest path problems by dynamic programming. *Operations Research*, pages 1381–1384, 1984.

G. Chen, B.M. Znosko, X. Jiao, and D.H. Turner. Factors affecting thermodynamic stabilities of rna 3× 3 internal loops. *Biochemistry*, 43(40):12865–12876, 2004. doi: 10.1021/bi049168d. URL `http://dx.doi.org/10.1021/bi049168d`.

F. Costa and K. De Grave. Fast neighborhood subgraph pairwise distance kernel. In *Proceedings of the 26th International Conference on Machine Learning*, pages 255–262, 2010. URL `https://lirias.kuleuven.be/handle/123456789/267297`.

V. De Fonzo, F. Aluffi-Pentini, and V. Parisi. Hidden markov models in bioinformatics. *Current Bioinformatics*, 2(1):49–61, 2007. URL `http://books.google.de/books?hl=en&lr=&id=qgtSyjFQ9GAC&oi=fnd&pg=PR15&ots=bNmK78ubVx&sig=ae8MhuqlWyPTMiEfZnr9sUuQgaM&redir_esc=y#v=onepage&q&f=false`.

K.E. Deigana, T.W. Lia, D.H. Mathewsb, and K.M. Weeksa. Accurate shape-directed rna structure determination. *PNAS*, 106(1):97–102, 2009. doi: 10.1073/pnas.0806929106. URL `http://dx.doi.org/10.1073/pnas.0806929106`.

Y. Ding and C.E. Lawrence. A statistical sampling algorithm for rna secondary structure prediction. *Nucleic acids research*, 31(24):7280–7301, 2003. doi: 10.1093/nar/gkg938. URL `http://dx.doi.org/10.1093/nar/gkg938`.

C.B. Do, D.A. Woods, and S. Batzoglou. Contrafold: Rna secondary structure prediction without physics-based models. *Bioinformatics*, 22(14):e90–e98, 2006. doi: 10.1093/bioinformatics/btl246. URL `http://dx.doi.org/10.1093/bioinformatics/btl246`.

K. Doshi, J. Cannone, C. Cobaugh, and R. Gutell. Evaluation of the suitability of free-energy minimization using nearest-neighbor energy parameters for rna secondary structure prediction. *BMC bioinformatics*, 5(1):105, 2004. doi: 10.1186/1471-2105-5-105. URL `http://dx.doi.org/10.1186/1471-2105-5-105`.

R. Durbin. *Biological sequence analysis: Probabilistic models of proteins and nucleic acids*. Cambridge Univ Pr, 1998. doi: 10.1017/CBO9780511790492. URL http://dx.doi.org/10.1017/CBO9780511790492.

P.P. Gardner, J. Daub, J. Tate, B.L. Moore, I.H. Osuch, S. Griffiths-Jones, R.D. Finn, E.P. Nawrocki, D.L. Kolbe, S.R. Eddy, et al. Rfam: Wikipedia, clans and the "decimal" release. *Nucleic acids research*, 39(suppl 1):D141, 2011. doi: 10.1093/nar/gkq1129. URL http://dx.doi.org/10.1093/nar/gkq1129.

S. Griffiths-Jones, S. Moxon, M. Marshall, A. Khanna, S.R. Eddy, and A. Bateman. Rfam: annotating non-coding rnas in complete genomes. *Nucleic acids research*, 33(suppl 1):D121, 2005. doi: 10.1093/nar/gki081. URL http://dx.doi.org/10.1093/nar/gki081.

J.L. Gross and J. Yellen. *Handbook of graph theory*. CRC, 2004.

D. Haussler. Convolution kernels on discrete structures. 1999. doi: 10.1007/s10863-011-9338-7. URL http://dx.doi.org/10.1007/s10863-011-9338-7.

IL Hofacker, W. Fontana, PF Stadler, LS Bonhoeffer, and M. Tacker. Fast folding and comparison of rna secondary structures. *Monatshefte f
"ur Chemie*, 125:167–188, 1994. doi: 10.1007/BF00818163. URL http://dx.doi.org/10.1007/BF00818163.

S. Janssen and R. Giegerich. Faster computation of exact rna shape probabilities. *Bioinformatics*, 26(5):632–639, 2010. doi: 10.1093/bioinformatics/btq014. URL http://dx.doi.org/10.1093/bioinformatics/btq014.

D.H. Mathews and D.H. Turner. Prediction of rna secondary structure by free energy minimization. *Current opinion in structural biology*, 16(3):270–278, 2006. doi: 10.1016/j.sbi.2006.05.010. URL http://dx.doi.org/10.1016/j.sbi.2006.05.010.

D.H. Mathews, M.D. Disney, J.L. Childs, S.J. Schroeder, M. Zuker, and D.H. Turner. Incorporating chemical modification constraints into a dynamic programming algorithm for prediction of rna secondary structure. *PNAS*, 101(19):7287–7292, 2004. doi: 10.1073/pnas.0401799101. URL http://dx.doi.org/10.1073/pnas.0401799101.

M.L. Metzker. Emerging technologies in dna sequencing. *Genome research*, 15(12):1767–1776, 2005. doi: 10.1101/gr.3770505. URL http://dx.doi.org/10.1101/gr.3770505.

T.F. Smith and M.S. Waterman. Identification of common molecular subsequences. *Journal of Molecular Biology*, 147(1):195 – 197, 1981. ISSN 0022-2836. doi: 10.1016/0022-2836(81)90087-5. URL http://www.sciencedirect.com/science/article/pii/0022283681900875.

R.J. Taft, M. Pheasant, and J.S. Mattick. The relationship between non-protein-coding dna and eukaryotic complexity. *Bioessays*, 29(3):288–299, 2007. doi: 10.1002/bies.20544. URL http://dx.doi.org710.1002/bies.20544.

Ryan J Taft, Ken C Pang, Timothy R Mercer, Marcel Dinger, and John S Mattick. Non-coding rnas: regulators of disease. *The Journal of Pathology*, 220(2):126–139, 2010. ISSN 1096-9896. doi: 10.1002/path.2638. URL `http://dx.doi.org/10.1002/path.2638`.

B. Voß, R. Giegerich, and M. Rehmsmeier. Complete probabilistic analysis of rna shapes. *BMC biology*, 4(1):5, 2006. doi: 10.1186/1741-7007-4-5. URL `http://dx.doi.org/10.1186/1741-7007-4-5`.

Wiipedia. List of RNA structure prediction software, 2012a. URL `http://en.wikipedia.org/wiki/List_of_RNA_structure_prediction_software`. [Online; accessed 12-Feb-2012].

Wiipedia. Stockholm Format on WiKi, 2012b. URL `http://en.wikipedia.org/wiki/Stockholm_format`. [Online; accessed 8-Feb-2012].

S. WUCHTY, W. FONTANA, IL HOFACKER, and P. SCHUSTER. Complete suboptimal folding of rna and the stability of secondary structures. *Biopolymers*, 49(2):145–165, 1999. doi: 10.1002/(SICI)1097-0282(199902)49:2<145::AID-BIP4>3.0.CO;2-G. URL `http://dx.doi.org/10.1002/(SICI)1097-0282(199902)49:2<145::AID-BIP4>3.0.CO;2-G`.

X. Yan and J. Han. gspan: Graph-based substructure pattern mining. *Order A Journal On The Theory Of Ordered Sets And Its Applications*, 2:721–724, 2002. doi: 10.1109/ICDM.2002.1184038. URL `http://dx.doi.org/10.1109/ICDM.2002.1184038`.

B.J. Yoon and PP Vaidynathan. Hmm with auxiliary memory: a new tool for modeling rna structures. In *Signals, Systems and Computers, 2004. Conference Record of the Thirty-Eighth Asilomar Conference on*, volume 2, pages 1651–1655. IEEE, 2004. doi: 10.1109/ACSSC.2004.1399438. URL `http://dx.doi.org/10.1109/ACSSC.2004.1399438`.

M. Zuker and D. Sankoff. Rna secondary structures and their prediction. *Bulletin of Mathematical Biology*, 46(4):591–621, 1984. doi: 10.1007/BF02459506. URL `http://dx.doi.org/10.1007/BF02459506`.

M. Zuker and P. Stiegler. Optimal computer folding of large rna sequences using thermodynamics and auxiliary information. *Nucleic acids research*, 9(1):133, 1981. doi: 10.1093/nar/9.1.133. URL `http://dx.doi.org/10.1093/nar/9.1.133`.

List of Figures

Nomenklatur

DBN it is Dot-bracket notation for more see subsection 3.1.1

DP it is Dot-Parentheses same as Dot-Bracket notation for more see subsection 3.1.1

dynamic programming is a method for solving complex problems by breaking them down into simpler subproblems. It is applicable to problems exhibiting the properties of overlapping subproblems which are only slightly smaller and optimal substructure (described below). When applicable, the method takes far less time than naive methods.

Global alignment Global alignments, which attempt to align every residue in every sequence, are most useful when the sequences in the query set are similar and of roughly equal size. (This does not mean global alignments cannot end in gaps.) A general global alignment technique is the Needleman-Wunsch algorithm, which is based on dynamic programming

homomorphisms In abstract algebra, a homomorphism is a structure-preserving map between two algebraic structures (such as groups, rings, or vector spaces)

MFE Minimum free energy

nucleotide Nucleotide is the basic building block of nucleic acids which has three components a nitrogenous base, a sugar and a phosphate.

pairwise alignment Pairwise sequence alignment methods are used to find the best-matching piecewise (local) or global alignments of two query sequences. Pairwise alignments can only be used between two sequences at a time, but they are efficient to calculate and are often used for methods that do not require extreme precision (such as searching a database for sequences with high similarity to a query). The three primary methods of producing pairwise alignments are dot-matrix methods, dynamic programming, and word methods; however, multiple sequence alignment techniques can also align pairs of sequences.

Pseudo-knots When in an *RNA structure* at least consisting of two stem loop structures in which half of the one stem is intercalated between the two halves of another stem then it is called as pseudoknot

SCFG A stochastic context-free grammar (SCFG; also probabilistic context-free grammar, PCFG) is a context-free grammar in which each production is augmented with a probability. The probability of a derivation (parse) is then the product of the probabilities of the productions used in that derivation; thus some derivations are more consistent with the stochastic grammar than others. SCFGs extend context-free grammars in the same way that hidden Markov models extended regular grammars.

states state is a particular set of instructions that will be executed in response to the machine's input. The state can be thought of as analogous to a practical computer's main memory, speaks about the behavior of the system.