Computational methods for accurate and efficient identification of noncoding RNAs

September 2012

Yutaka Saito

A Thesis for the Degree of Ph.D. in Science

Computational methods for accurate and efficient identification of noncoding RNAs

September 2012

Graduate School of Science and Technology
Keio University

Yutaka Saito

Contents

Abbreviation

AUC:	area under the ROC curve
BPLA:	base-pairing profile local alignment
CM:	covariance model
DP:	dynamic programming
FN:	false negative
FP:	false positive
HMM:	hidden Markov model
LA:	local alignment
LSH:	locality-sensitive hashing
RBF:	radial basis function
ROC:	receiver operating characteristic
SCI:	structure conservation index
SVM:	support vector machine
SW:	Smith-Waterman
TN:	true negative
TP:	true positive
UCSC:	University of California, Santa Cruz
WPGMA:	weighted pair-group method with arithmetic mean
IRES:	internal ribosome entry site
lncRNA:	long ncRNA
LSU:	large subunit
miRNA:	micro RNA
mRNA:	messenger RNA
NAT:	natural antisense transcript
ncRNA:	noncoding RNA
PAR:	promoter associated RNA
piRNA:	Piwi-interacting RNA
rasiRNA:	repeat associated siRNA
rRNA:	ribosomal RNA
sdRNA:	sno-derived RNAs
siRNA:	short interfering RNA
snoRNA:	small nucleolar RNA
snRNA:	small nuclear RNA
SRP:	signal recognition particle
SSU:	small subunit
svRNA:	small vault RNA
tmRNA:	transfer-messenger RNA
TR:	telomerase RNA
tRF:	tRNA-derived RNA fragment
tRNA:	transfer RNA

Chapter 1

Introduction

Genome projects have revealed a striking fact that the vast majority of genomic sequences in higher organisms do not code for proteins (Siepel *et al.*, 2005). The fraction of noncoding regions increases along with the complexity of organisms, reaching as high as 98% in humans (Taft *et al.*, 2007). *Noncoding RNAs* (ncRNAs) are transcripts from the genes present in these regions, and are revolutionizing the traditional "protein-centered" view of functional genomics (Hüttenhofer *et al.*, 2005). Known ncRNA families include ribosomal RNAs (rRNAs), transfer RNAs (tRNAs), micro RNAs (miRNAs) which mediate post-transcriptional regulation (Filipowicz *et al.*, 2008), small nucleolar RNAs (snoRNAs) which guide modification of other RNAs (Kiss, 2001), and riboswitches which respond to changes in metabolite concentrations (Dambach and Winkler, 2009). Our knowledge of ncRNAs is still in its infancy when compared to that of proteins. Figure 1.1 shows the accumulation of database entries for ncRNAs in comparison to those for proteins. The Rfam database for ncRNAs (Gardner *et al.*, 2011) was started from 2002, which is much later than 1996 when the Pfam database for proteins (Punta *et al.*, 2012) was founded. Currently, the Rfam database collects only 3×10^6 ncRNA sequences as members in known families, while the Pfam database amounts to 13×10^6 protein sequences (Figure 1.1a). In addition, only 1973 ncRNA families are established in the Rfam database, which is much less than 13672 protein families in the Pfam database (Figure 1.1b). To address these issues, the development of computational methods for accurate and efficient identification of ncRNAs has been a major goal of bioinformatics (Eddy, 2002).

In this dissertation, we present two computational methods that solve different problems for the identification of ncRNAs. First, we propose a method that predicts whether an input RNA is a new member of a known ncRNA family (Saito *et al.*, 2010). The method is useful to increase the number of member sequences in known families, addressing the issue represented in Figure 1.1a. Second, we propose a method that finds candidates of novel ncRNA families from a set of unannotated RNAs (Saito *et al.*, 2011). The method is useful to increase the number of established families, addressing the issue represented in Figure 1.1b. The relationship of the two methods in the identification of ncRNAs are shown in Figure 1.2. The first method performs the annotation of genomes or transcriptomes in which each sequence is classified into one of known families. This process leaves a set of unannotated sequences that cannot be confidently classified into any of known families. The second method takes these unannotated sequences as input, and finds candidates of novel families.

This dissertation is organized as follows. In the remainder of this chapter, we introduce some basic knowledge about ncRNAs, and briefly explain the originality of our study from the perspective of RNA informatics. In Chapter 2, a method for predicting new members of known families are described. In Chapter 3, a method for

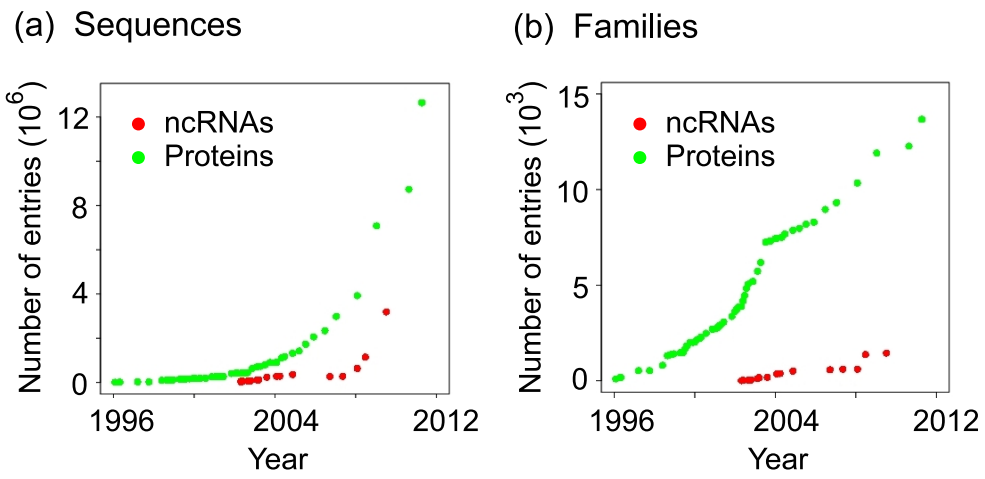


Figure 1.1 The accumulation of database entries for ncRNAs in comparison to those for proteins. For each version of a database, the number of entries are plotted against its release date. The data are obtained from the Rfam database for ncRNAs, and the Pfam database for proteins. (a) The accumulation in the number of member sequences in known families. (b) The accumulation in the number of established families.

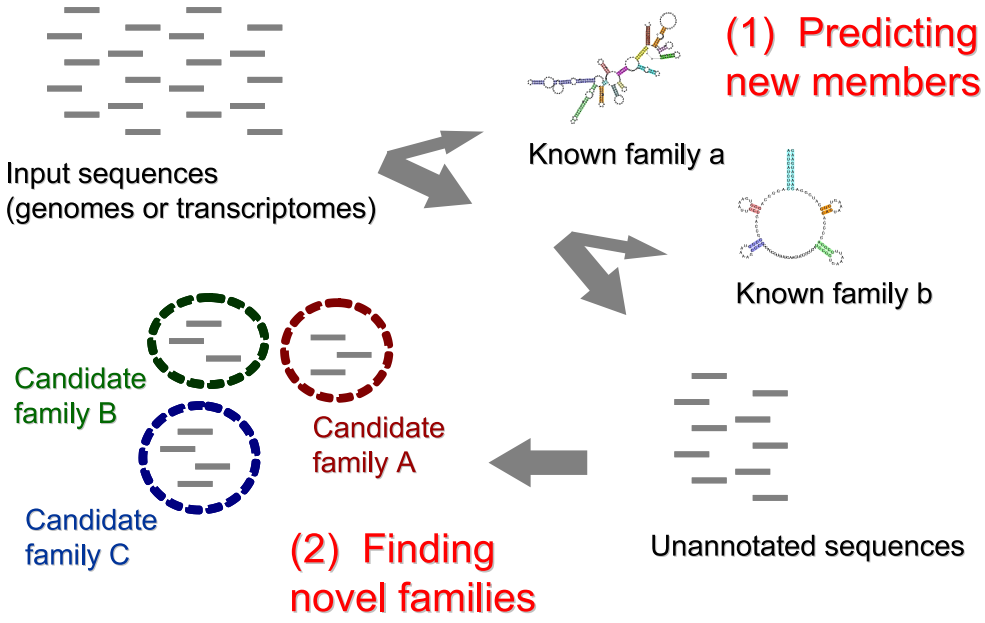


Figure 1.2 The relationship of the two methods proposed for the identification of ncRNAs. The first method identifies new members of known families from a set of sequences in genomes or transcriptomes. The second method identifies candidates of novel families from a set of sequences which do not belong to any of known families. The two methods can be combined to constitute a framework for the identification of ncRNAs.

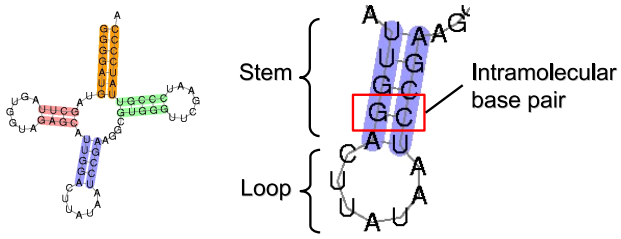


Figure 1.3 Typical example of secndary structures for tRNAs. The left-hand figure shows an entire secondary structure of a tRNA molecule. The right-hand figure magnifies one stem loop structure which has base pairs colored in blue.

finding candidates of novel families are described. In Chapter 4, we summarize our study and discuss future prospects.

1.1 Secondary structures of ncRNAs

In this section, we introduce a variety of ncRNA families, focusing on the relationship between cellular functions and secondary structures. The cellular functions of ncRNAs are often associated with their secondary structures formed by intramolecular base pairs (Eddy, 2001). Figure 1.3 shows a typical example of secondary structures for tRNAs. The left-hand figure shows a secondary structure in which a tRNA molecules folds back itself by base pairs. The structure consists of four substructures, each of which has base pairs shown in a different color. The right-hand figure magnifies one substructure with blue-colored base pairs. Such a substructure is called a stem-loop structure, where a stretch of stacked base pairs is called a stem, while an unpaired region closed by a stem is called a loop. Each ncRNA family in the Rfam database is defined by its own secondary structure conserved through the evolution.

1.1.1 rRNAs

The rRNAs serve as a components of the ribosome, forming a complex with ribosomal proteins. The both of small subunit (SSU) and large subunit (LSU) rRNAs have their own secondary structure which exhibits a strong conservation. Figure 1.4a shows the secondary structure of SSU rRNAs in bacteria. Structural domains of SSU rRNAs are often referred to as variable regions because nucleotide sequences in these region are frequently mutated. However, these domains are still conserved in terms of their secondary structure due to the co-mutation of pairing nucleotides. Several studies have suggested that the phylogenetic reconstruction based on rRNAs can be improved when incorporating the secondary structure information (Mallatt and Winchell, 2007; Stocsits *et al.*, 2009).

1.1.2 tRNAs

The tRNAs carry a specific type of amino acid, depending on their anticodons, to the ribosome during translation. The conserved secondary structure of tRNAs is known as a clover-leaf shape shown in Figure 1.4b. An anticodon is located in the loop region of the anticodon arm (a stem-loop structure magnified in Figure 1.3). It has been suggested that the the structure of tRNAs, together with those of rRNAs and ribosomal proteins, play an important role for codon-anticodon pairing (Ogle *et al.*, 2003).

1.1.3 miRNAs

The miRNAs regulate the expression of messenger RNAs (mRNAs) in a post-transcriptional manner based on the interaction via a complementary sequence (Filipowicz *et al.*, 2008). There have been extensive studies on a large variety of processing pathways for the maturation of miRNAs, including those for canonical miRNAs, canonical intronic miRNAs, and non-canonical intronic miRNAs (for review, see (Kim *et al.*, 2009)). Here, we only introduce the pathway for canonical miRNAs focusing on its relationship to secondary structures. A primary transcript exhibits one or more stem-loop structures, each of which corresponds to one mature miRNA. In a cell nucleus, Drosha and Pasha proteins recognize and cleave each stem-loop structure, producing a miRNA precursor (Figure 1.4c). After the export to cytoplasm, the stem of a miRNA precursor is recognized by Dicer protein, and cleaved as a double-stranded RNAs. Then, either side of a double-stranded RNA is loaded into Ago protein as a mature miRNA (with a preference possibly depending on nucleotide sequences).

1.1.4 snoRNAs

The snoRNAs localize in a nucleolus, and guide the modification of other (usually ribosomal) RNAs (Kiss, 2001). They are divided into two categories, called C/D snoRNAs and H/ACA snoRNAs, which conduct 2'-O-methylation and pseudourydilation of rRNAs, respectively. The conserved secondary structures for C/D and H/ACA snoRNAs are shown in Figure 1.4d and Figure 1.4e, respectively. C/D and H/ACA snoRNAs have sequence motifs, namely C/D box and H/ACA box motifs, at the specific positions in the context of their secondary structures. Moreover, the bulge loops in their secondary structures contain the sequences complementary to target rRNAs, and the modification to rRNAs is occurred at the specific residue in these regions.

1.1.5 Riboswitches

Riboswitches are ncRNAs which modulate the gene expression in response to changes in metabolite concentrations (Dambach and Winkler, 2009). Usually, riboswitches

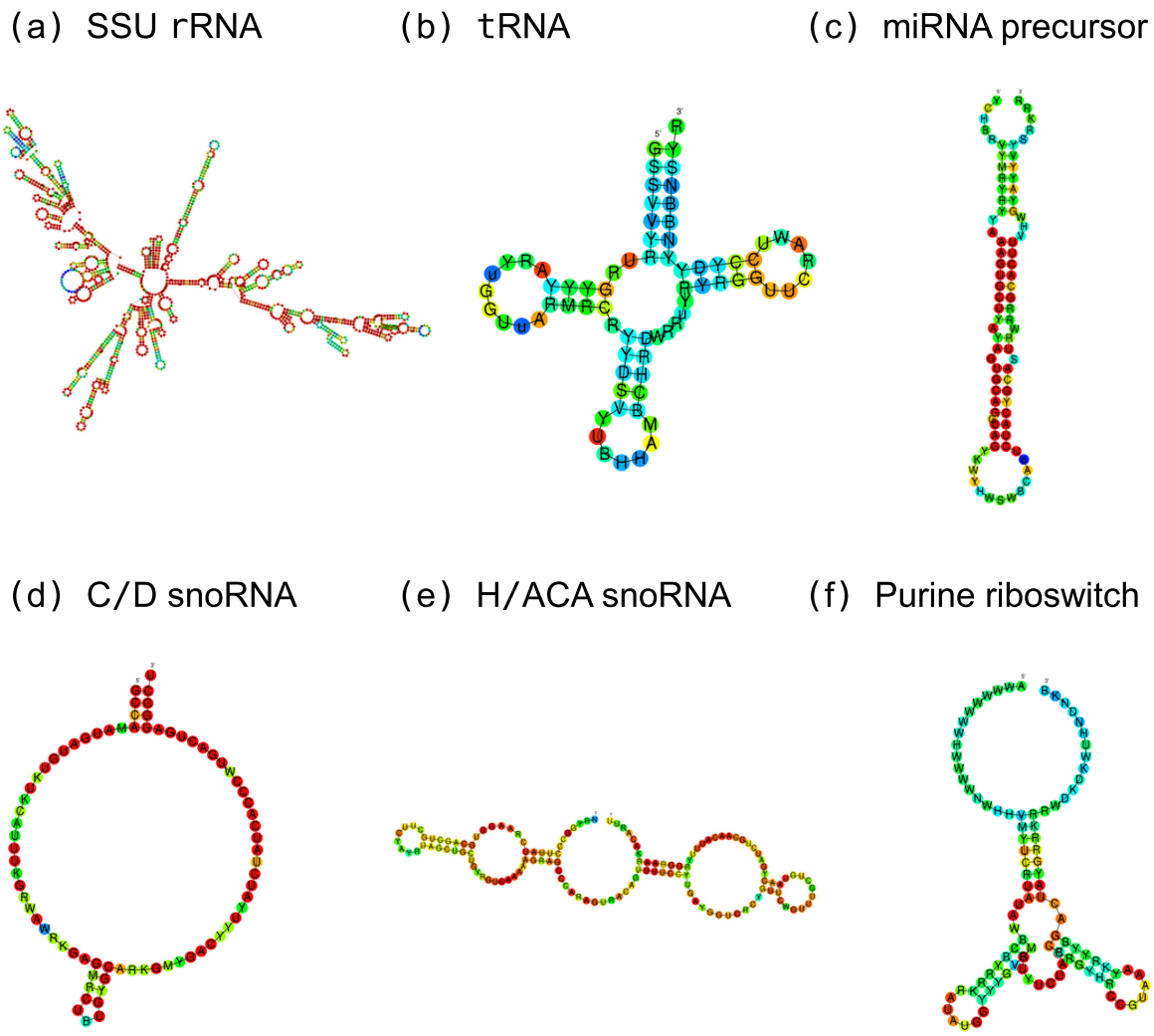


Figure 1.4 A variety of ncRNA families and their characteristic secondary structure. For each ncRNA family, the consensus secondary structure registered in the Rfam database is shown. (a) SSU rRNA in bacteria (Rfam accession RF00177). (b) tRNA (Rfam accession RF00005). (c) miRNA precursor mir-17 (Rfam accession RF00051). (d) C/D snoRNA SNORD83 (Rfam accession RF00137). (e) H/ACA snoRNA SNORA16 (Rfam accession RF00190). (f) Purine riboswitch (Rfam accession RF00167).

are metabolite-binding domains of bacterial mRNAs, and change their secondary structures by the binding so that the biosynthesis of the encoded proteins is regulated. Figure 1.4f shows an example of secondary structures for purine riboswitches (when not bound by purines). Other types of riboswitches include those for responding to temperature changes, and those for modulating alternative splicing in fungi (Serganov and Patel, 2007).

1.1.6 Other structured RNAs

There are many other ncRNAs families which can be well-characterized by their conserved secondary structures.

Some structured RNAs are components of huge complexes formed by ncRNAs and proteins. They include signal recognition particle (SRP) RNAs in SRPs which target membrane proteins to the endoplasmic reticulum (Rosenblad *et al.*, 2009), vault RNAs in vault particles which are involved in drug resistance (Stadler *et al.*, 2009), Y RNAs in Ro particles which conduct the quality control of rRNAs (Perreault *et al.*, 2007), and small nuclear RNAs (snRNAs) such as spliceosomal RNAs in spliceosomes. Telomerase RNAs (TRs) form a complex with telomere reverse transcriptase, and serve as a template for telomere elongation (Harley, 2008).

There exist a wide range of ncRNAs which are derived from mature ncRNAs, and exhibit functions other than the original ncRNAs. Some of these ncRNAs are cleaved from stem-loop structures present in larger secondary structures, and show miRNA-like regulatory functions. Examples are sno-derived RNAs (sdRNAs) cleaved from snoRNAs (Taft *et al.*, 2009a), and small vault RNAs cleaved from vault RNAs (Persson *et al.*, 2009).

In bacteria and retroviruses, a number of *cis*-acting elements are found within mRNAs. They include internal ribosome entry sites (IRESs) which enable the 5' cap-independent translation initiation (Lukavsky, 2009), transport elements which allow to export intact RNAs from a cell nucleus without being processed by the RNA splicing machinery (Smulevitch *et al.*, 2005). Transfer-messenger RNAs (tmRNAs) resemble both of tRNAs and mRNAs, and rescue ribosomes which accidentally stall during the translation of degraded mRNAs (Dulebohn *et al.*, 2007).

Prokaryotes have ncRNAs that have functional analogies to eukaryotic miRNAs, even though the proteins for processing these ncRNAs are non-homologous those for miRNAs (Majdalani *et al.*, 2005). In bacteria, small regulatory RNAs are mediated by the RNA chaperone Hfq (Aiba, 2007).

1.1.7 Non-structured RNAs

Although most ncRNA families are defined by their secondary structures, some ncRNAs do not exhibit significant secondary structures. They include small RNAs which do not exhibit stem-loop structures such as PIWI-interacting RNAs (piRNAs), also known as repeat-associated short interfering RNAs (rasiRNAs) (Malone and Hannon, 2009). In addition, tRNA-derived RNA fragments (tRFs) show a cleavage pattern different from miRNA-like derived RNAs (Lee *et al.*, 2009). Moreover, long ncRNAs (lncRNAs) such as natural antisense transcripts (NATs) often lack significant secondary structures, and seem to function by the direct sense-antisense interaction rather than intramolecular base pairs (Lapidot and Pilpel, 2006). Promoter associated RNAs (PARs), including long and small RNAs, also lack secondary structure (Taft *et al.*, 2009b).

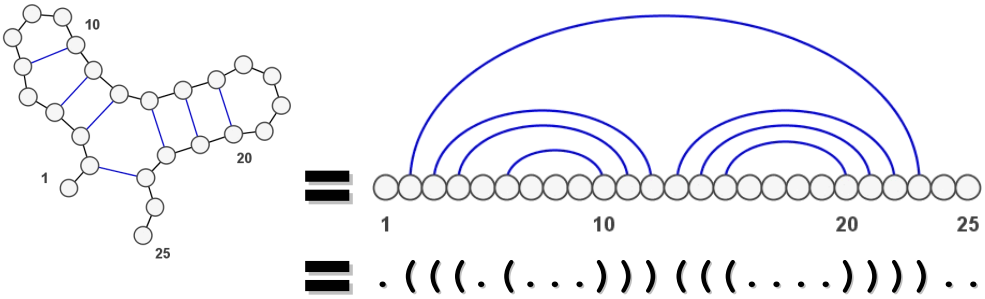


Figure 1.5 Graphical and string representations of a secondary structure. In the graphical representation, small circles represent residues in a nucleotide sequence, while blue lines represent base pairs among residues. In the string representation, a pair of left and right brackets represents a base pair, while dots represent unpaired residues.

As discussed later, the proposed methods in this dissertation assume that secondary structure information is useful to evaluate functional similarity of ncRNAs. Therefore, we cannot expect the proposed methods perform well on non-structured ncRNAs. We specifically address the identification of structured RNAs, and non-structured RNAs are out of the scope of our study.

1.2 RNA informatics

A secondary structure of an RNA molecule can be represented by string data along with its nucleotide sequences. Figure 1.5 shows a secondary structure of an RNA molecule using various notation schemes. First, we consider the most intuitive notation in which the RNA molcule folds back itself so that base-pairing positions connected by blue lines become close to each other. Next, we stretch the folded molecule into a straight line, while keeping the connection between base-pairing positions. This gives a notation in which base-pairing positions are represented by a set of nested arcs. Then, we place a pair of brackets in each of base-pairing positions, and a dot character in each of unpaired position. The resultant string representation is called a dot-bracket notation, which is commonly used in RNA informatics.

Some algorithms exist for extracting secondary structure information from nucleotide sequences using thermodynamic energy models (Zuker and Stiegler, 1981; McCaskill, 1990). This information, in addition to nucleotide sequences, can be exploited for the further information analysis of ncRNAs. There is a broad range of studies on RNA informatics, including structure-aware alignment (Sankoff, 1985), tertiary structure prediction (Parisien and Major, 2008), RNA-RNA interaction prediction (Kato *et al.*, 2010), and RNA-protein interaction prediction (Kazan *et al.*, 2010).

In the perspective of information analysis, we develop both of the two proposed methods as applications of *similarity search*, a fundamental task of biological sequence analysis. The originality of our studies is the design of similarity measures that

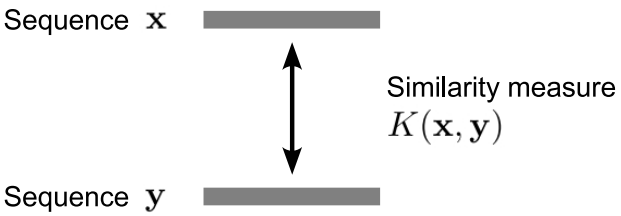


Figure 1.6 The simplest case of similarity search. Given a pair of sequences, $\mathbf{x}$ and $\mathbf{y}$, a similarity measure, $K(\mathbf{x}, \mathbf{y})$, is evaluated. The larger the similarity value is, the more likely the sequences belong to the same functional family.

greatly improve the performance of similarity search for ncRNAs. We need to design a similarity measure that compares a pair of RNAs by utilizing features related to their biological functions. For this purpose, we employ secondary structure information in addition to nucleotide sequences.

1.3 Similarity search and its applications

Figure 1.6 illustrates the simplest case of similarity search. Given a pair of sequences, $\mathbf{x}$ and $\mathbf{y}$, it is evaluated how likely they share a common biological function by using a certain similarity measure, $K(\mathbf{x}, \mathbf{y})$. The larger the similarity value is, the more likely the sequences belong to the same functional family. This process can be combined with some statistical frameworks to develop a computational method for a specific problem.

In our studies, we employ support vector machines (SVMs) (Boser *et al.*, 1992) for predicting new members of known ncRNA families, and clustering (Sokal and Michener, 1958) for finding novel ncRNA families.

1.3.1 Predicting new members of known families

We can consider an input sequence as a new member of a known family if the sequence is similar to existing members in the family. Therefore, we evaluate a similarity measure between the sequence and each of family members. The difficulty here is that we need to integrate multiple similarity values into one criterion for prediction. In addition, we must discriminate true similarity to family members from false similarity to non-members occured by chance. This problem is expressed by the following formula:

$$f(\mathbf{x}) = \sum_i \lambda_i K(\mathbf{x}, \mathbf{y}_+^i) - \sum_j \lambda_j K(\mathbf{x}, \mathbf{y}_-^j), \qquad \text{(Eq. 1.1)}$$

where $\mathbf{x}$ is an input sequence, $\{\mathbf{y}_+^i\}$ is a set of family members, $\{\mathbf{y}_-^j\}$ is a set of non-members, and λ_i and λ_j are weights for the contributions of family members and non-members, respectively. In (Eq. 1.1), the first term evaluates true similarity to family members, while the second term evaluates false similarity to non-members.

Thus, we predict that an input sequence $\mathbf{x}$ is a new member if $f(\mathbf{x}) \geq 0$.

To determine λ_i and λ_j in (Eq. 1.1), we employ a statistical framework called SVMs (Boser *et al.*, 1992). Figure 1.7 illustrates SVMs for predicting new members of know families. In a training phase, SVMs use family members $\{\mathbf{y}^i_+\}$ as positive samples, and non-members $\{\mathbf{y}^j_-\}$ as negative samples. These samples are mapped into a space where relative positions are consistent with similarity relationship defined by K (Figure 1.7a). (The similar samples, in terms of K, are mapped to a neighborhood in a space. SVMs determine a hyperplane in the space so that it can discriminate family members and non-members. This is equivalent to optimize λ_i and λ_j so that each of family members can take $f(\mathbf{y}^i_+) \geq 0$, and each of non-members can take $f(\mathbf{y}^j_-) < 0$. In a test phase, SVMs predict that an input sequence $\mathbf{x}$ is a new member if $\mathbf{x}$ is mapped into the family side of the space, *i.e.* $f(\mathbf{x}) \geq 0$ (Figure 1.7b). For a more precise formulation of SVMs, see (Boser *et al.*, 1992) and (Vapnik, 1998).

In the context of SVMs, a similarity measure, K, is called a *kernel function*. The performance of an SVM classifier depends critically on the design of a kernel function since it defines relative positions in the space.

1.3.2 Finding novel families

We can find a candidate of novel family from a set of unannotated sequences by detecting subsets in which sequences are similar to each other. Such a problem is called clustering, and requires to evaluate a similarity measure among all pairs in a given set.

Specifically, we employ hierarchical clustering by the weighted pair-group method with arithmetic mean (WPGMA) algorithm (Sokal and Michener, 1958). Figure 1.8 illustrates a clustering procedure for finding novel families. Given a set of unannotated sequences, we compute an all-against-all similarity matrix using K, and derive the distance matrix by one minus the similarity (assuming a similarity value is normalized to range from one to zero). The WPGMA algorithm constructs a cluster tree whose leaves are sequences, and branches (edges) represent the distance among sequences. We can obtain candidate families from the cluster tree by cutting the branches at a distance threshold. For a more precise description of the WPGMA algorithm, see (Sokal and Michener, 1958).

The performance of clustering methods depends critically on the design of a similarity measure, K. In particular, if a similarity is inaccurate, a cluster tree becomes unclear, and requires manual inspection to detect novel families.

1.4 Designing a similarity measures for ncRNAs

In this section, we clarify the difficulty in designing a similarity measure for ncRNAs, and briefly explain the originality of our study.

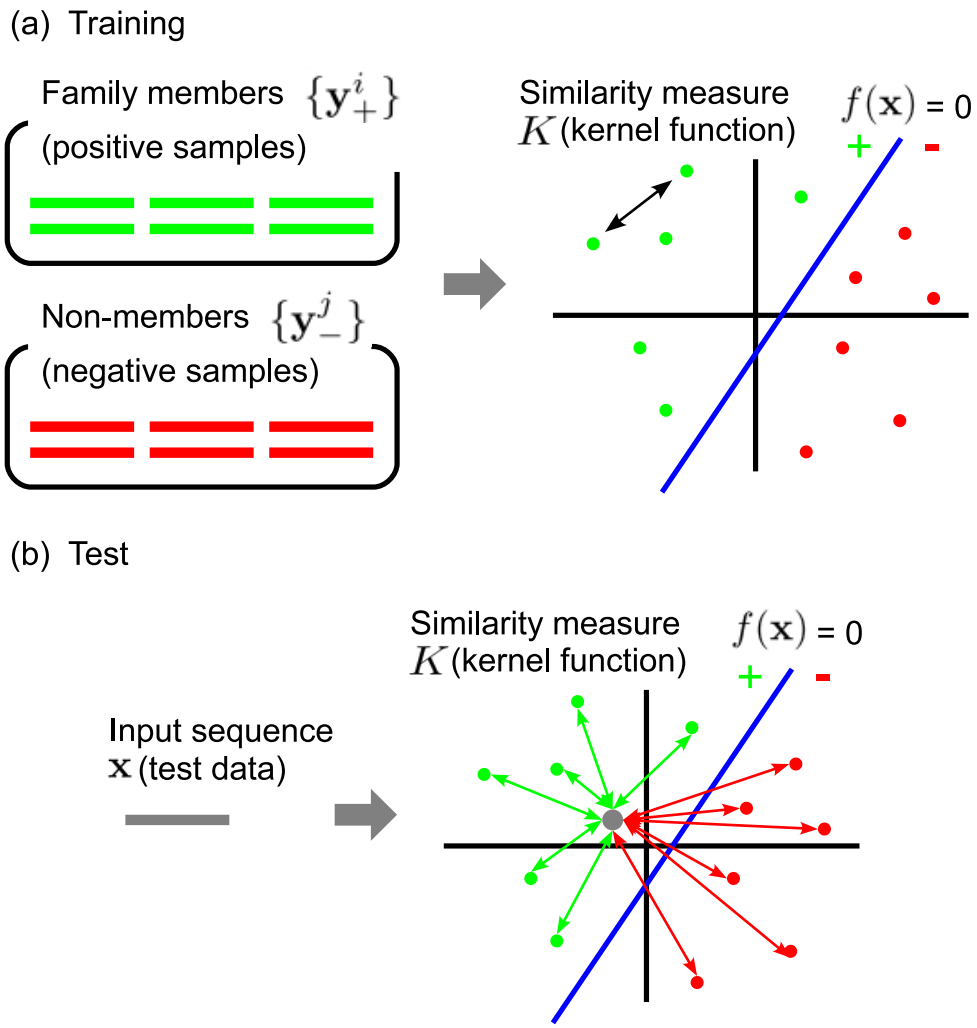


Figure 1.7 Schematic diagram of SVMs for predicting a new member of a known family. (a) In a training phase, family members $\{\mathbf{y}_+^i\}$ and non-members $\{\mathbf{y}_-^j\}$ are used as positive samples and negative samples, respectively. These samples are mapped into a space where relative positions are consistent with similarity relationship defined by K. SVMs determine a hyperplane in the space so that it can discriminate family members and non-members. (b) In a test phase, SVMs predict that an input sequence $\mathbf{x}$ is a new member or not, depending on whether $\mathbf{x}$ is mapped into the family side of the space,

1.4.1 Similarity of nucleotide sequences

We can measure the similarity between two nucleotide sequences by pairwise alignment using the Smith-Waterman (SW) algorithm (Smith and Waterman, 1981). The SW algorithm calculates the similarity of entire sequences based on a scoring function, $S_{\mathbf{xy}}(i, j)$, which measures the similarity between the i-th position in $\mathbf{x}$ and the j-th position in $\mathbf{y}$ (Figure 1.9). If we do not consider secondary structure

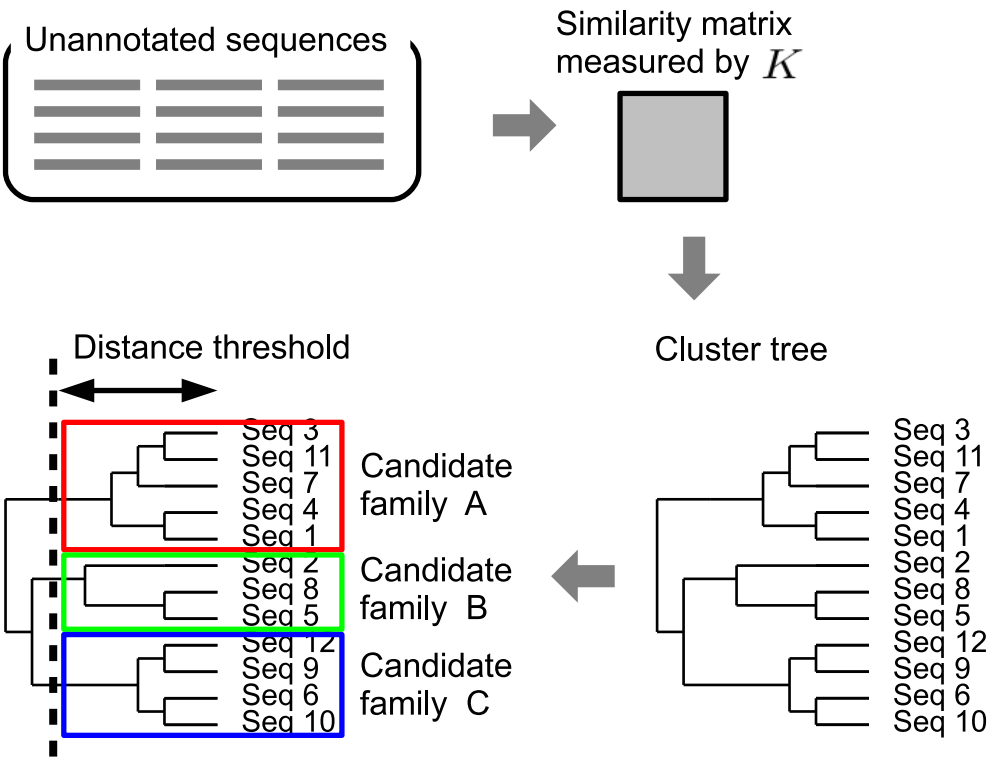


Figure 1.8 Schematic diagram of clustering procedures for finding candidates of novel families. For a set of unannotated sequences, an all-against-all similarity matrix is computed by using K, the corresponding distance matrix is derived. A cluster tree is constructed so that the lengths of branches (edges) represents the distance among sequences. Candidate families are obtained from the cluster tree by cutting the branches at a distance threshold.

information, the scoring function is simply a similarity measure for nucleotide characters, $\{\mathtt{A}, \mathtt{C}, \mathtt{G}, \mathtt{U}\}$, *i.e.* a substitution matrix. We can employ a substitution matrix such as the RIBOSUM matrix (Klein and Eddy, 2003), which corresponds to the BLOSUM matrix for amino acid characters.

In the SW algorithm, the scoring function, $S_{\mathbf{xy}}(i, j)$, needs to be evaluated for $O(|\mathbf{x}||\mathbf{y}|)$ combinations of positions because i and j can take $|\mathbf{x}|$ and $|\mathbf{y}|$ possible values, respectively. Consequently, the SW algorithm require the computational cost in $O(|\mathbf{x}||\mathbf{y}|)$.

1.4.2 Similarity of secondary structures

To measure the similarity of secondary structures, a straightforward approach is to consider the scoring function between two base pairs rather than two residues. Figure 1.10a shows this situation. The scoring function, $S_{\mathbf{xy}}(i, v, j, w)$, measures the similarity between the base pair (i, v) in $\mathbf{x}$ and the base pair (j, w) in $\mathbf{y}$, incorporating secondary structure information. However, the scoring function needs to be evaluated

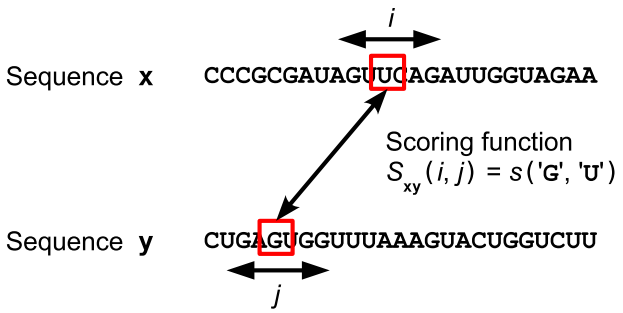


Figure 1.9 A scoring function that measures the similarity of nucleotide sequences. The scoring function needs to be evaluated for $O(|\mathbf{x}||\mathbf{y}|)$ combinations of positions because i and j can take $|\mathbf{x}|$ and $|\mathbf{y}|$ possible values, respectively.

for $O(|\mathbf{x}|^2|\mathbf{y}|^2)$ combination of positions because two variables exist for each of $\mathbf{x}$ and $\mathbf{y}$. This computation is usually prohibitive for practical applications.

One compromise is to ignore the partner of base pairs, and consider the scoring function between two residues as shown in Figure 1.10b. The scoring function can still incorporate secondary structure information observed from the one side of a base pair (*i.e.* the information that the residue is left or right side of a base pair, or unpaired). The scoring function needs to be evaluated only in $O(|\mathbf{x}||\mathbf{y}|)$ combinations of positions, which is the same order as the SW algorithm. However, this heuristics for the scoring function may degrade the accuracy of resultant similarity measures. Therefore, we compensate this approximation by incorporating additional information.

1.4.3 Profile information

In Chapter 2, we describe a new kernel function, called Profile BPLA kernel, which predicts ncRNAs from alignment data rather than single sequences. By utilizing the profile information of alignment data, the proposed kernel enables to calculate the accurate similarity between ncRNAs (Figure 1.11).

We achieve better accuracy than existing methods (Morita *et al.*, 2009; Washietl *et al.*, 2005; Gruber *et al.*, 2010; Sato *et al.*, 2008) for a wide range of families including miRNAs, snoRNAs and riboswitches. Furthermore, our method can keep its excellent performance under the practical condition where the quality of input alignment data is not necessarily high. We simulate errors in alignment data suggested by previous studies (Prakash and Tompa, 2007; Wang *et al.*, 2007; Torarinsson *et al.*, 2006, 2008), and evaluate to what extent the performance of prediction methods can be influenced. Our method is surprisingly robust against the errors even when existing methods are severely damaged.

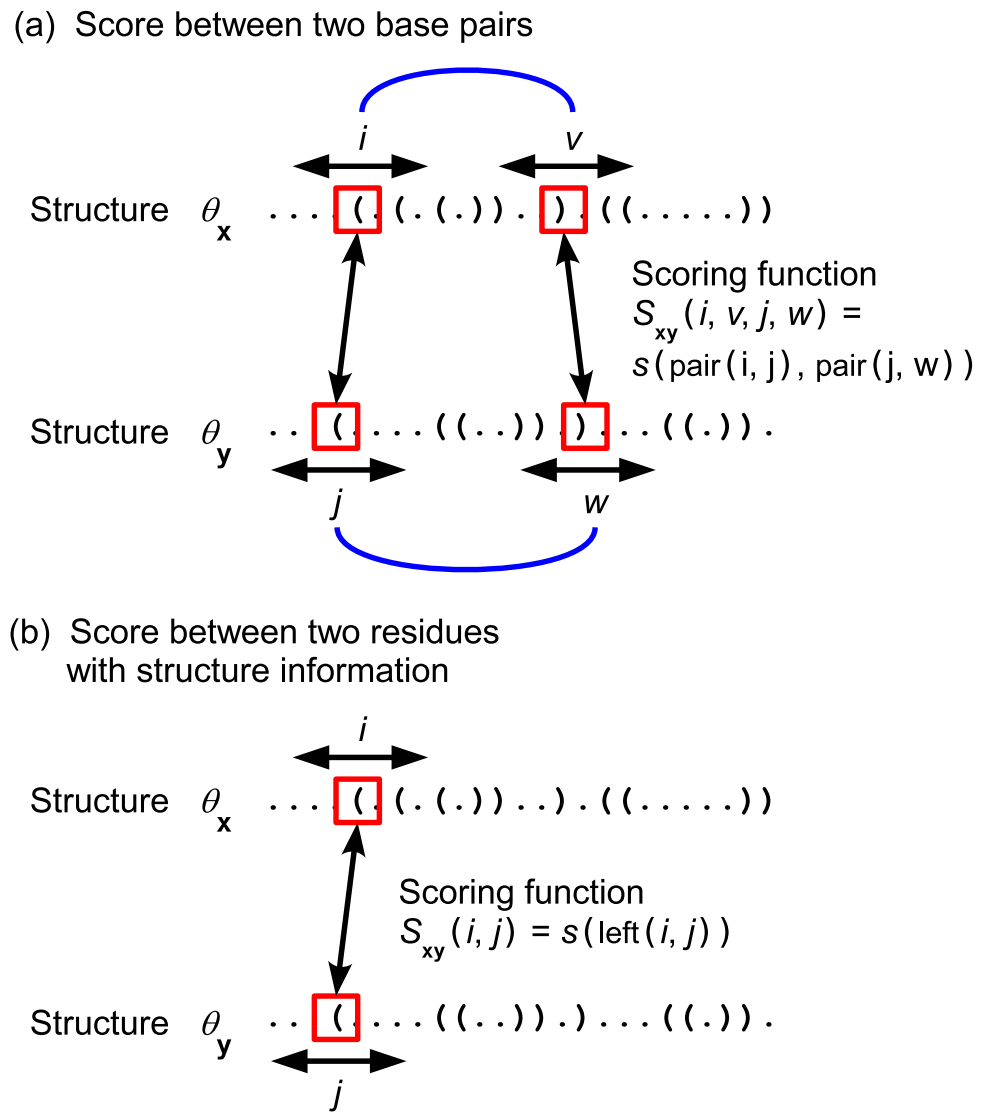


Figure 1.10 Scoring functions that measure the similarity of secondary structures. (a) A scoring function that measures the similarity between two base pairs needs to be evaluated for $O(|\mathbf{x}|^2|\mathbf{y}|^2)$ combinations of positions. This is too time-consuming. (b) A similarity measure is approximated so that it ignores the partner of base-pairing, but still utilizes secondary structure information in each position. The approximate similarity measure needs to be evaluated only in $O(|\mathbf{x}||\mathbf{y}|)$ combinations of positions.

1.4.4 Ensemble information

In Chapter 3, we describe a new similarity measure that utilizes the ensemble information involved when compairing ncRNAs. Our similarity measure incorporate *all possible* sequence alignments and *all possible* secondary structures predicted from a pair of RNAs (Figure 1.12).

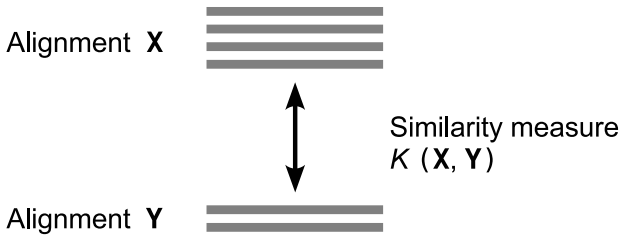


Figure 1.11 Extension of similarity search using profile information. A similarity measure is defined between alignment data, rather than single sequences.

We achieve the best balance between accuracy and efficiency among existing methods (Will *et al.*, 2007; Torarinsson *et al.*, 2007; Sato *et al.*, 2008). The improvement is especially remarkable when a family to be detected has a large diversity of member sequences. Our method can provide candidate families without manual inspection required by existing methods. Moreover, our method is about 1000 times as fast as previous state-of-the-art methods, making it more attractive for large-scale analysis.

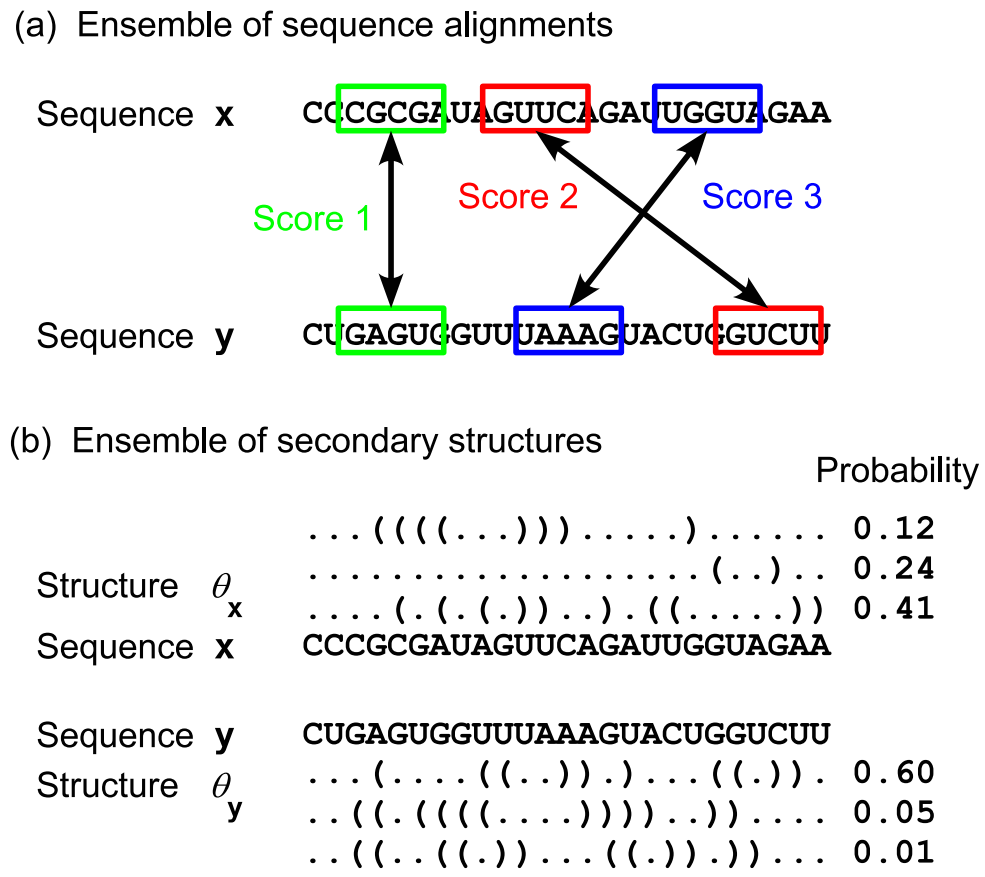


Figure 1.12 Ensemble information in the evaluation of a similarity measure. (a) Two sequences have a lot of possible sequence alignments, which have different alignment scores. (b) Each of two sequences has a lot of possible secondary structures, which have different probabilities.

Chapter 2

Robust and accurate prediction of noncoding RNAs from aligned sequences

In this chapter, we propose a method that predicts whether an input RNA is a new member of a known noncoding RNA (ncRNA) family (Saito *et al.*, 2010). This problem can be considered as an application of similarity search in which a similarity measure between a pair of RNAs is combined with support vector machines (SVMs) to discriminate family members from non-members. Thus, we aim to develop a similarity measure which is called a kernel function in the context of SVMs.

To measure the similarity between a pair of RNAs accurately, one common approach is to utilize the profile information contained in alignment data rather than single sequences. However, this strategy involves the possibility that the quality of input alignments can influence the performance of prediction methods. Therefore, the evaluation of the robustness against alignment errors is necessary as well as the development of accurate prediction methods.

We describe a new method, called Profile BPLA kernel, which predicts ncRNAs from alignment data in combination with SVMs. Profile BPLA kernel is an extension of *base-pairing profile local alignment* (BPLA) kernel which we previously developed for the prediction from single sequences. By utilizing the profile information of alignment data, the proposed kernel can achieve better accuracy than the original BPLA kernel. We show that Profile BPLA kernel outperforms the existing prediction methods which also utilize the profile information using the high-quality structural alignment dataset. In addition to these standard benchmark tests, we extensively evaluate the robustness of Profile BPLA kernel against errors in input alignments. We consider two different types of error: first, that all sequences in an alignment are actually ncRNAs but are aligned ignoring their secondary structures; second, that an alignment contains unrelated sequences which are not ncRNAs but still aligned. In both cases, the effects of errors on the performance of Profile BPLA kernel are surprisingly small. Especially for the latter case, we demonstrate that Profile BPLA kernel is more robust compared to the existing prediction methods.

2.1 Background

Reliable identification of ncRNAs is one of the major goals of recent computational biology (Eddy, 2002; Hüttenhofer *et al.*, 2005). To improve the reliability of predictions, many existing methods take an alignment as input rather than a single sequence (Backofen *et al.*, 2007). Alignment data provide the profile information of ncRNAs which is not evident from individual sequences; it can help to capture detailed features of primary sequences and secondary structures. Several prediction methods based on SVMs have been proposed with this respect, and shown to achieve

high accuracy (Washietl *et al.*, 2005; Gruber *et al.*, 2010; Sato *et al.*, 2008). Each method has its own kernel function which defines the similarity between a pair of alignment data and determines the accuracy of the SVM classifier. Washietl *et al.* (2005) and Gruber *et al.* (2010) have developed RNAz, which employs radial basis function (RBF) kernels to compute the similarity of feature vectors of alignment data. A major contribution to its prediction is made by the structure conservation index (SCI) based on thermodynamic energy models. This feature value assesses whether an alignment is structurally conserved by normalizing the minimum free energy of consensus secondary structures with the average of those for individual sequences. Sato *et al.* (2008) have developed Profile stem kernel as an extension of Stem kernel which was originally proposed for analyzing single sequences (Sakakibara *et al.*, 2007). The method calculates the similarity between a pair of alignment data by summing the substitution scores for all pairs of effective (highly probable) consensus stem structures.

In their studies, input alignments were assumed to be correct or at least not damaging to the accuracy of the prediction methods. However, it is not necessarily the case under the realistic conditions in genomic and transcriptomic screens. Since aligning genomic sequences is an error-prone process (Prakash and Tompa, 2007; Wang *et al.*, 2007), prediction methods have to deal with low-quality alignment data in most practical applications. For example, RNAz and Profile stem kernel utilize consensus secondary structures as the profile information, which are known to be degraded by the use of low-quality alignment data (Kiryu *et al.*, 2007). The previous studies have not fully evaluated to what extent the quality of input alignments can influence the performance of the prediction methods.

We can consider two different types of error in alignment data: first, that all sequences in an alignment are actually ncRNAs but are aligned ignoring their secondary structures (Type A); second, that an alignment contains unrelated sequences which are not ncRNAs but still aligned (Type B). Figure 2.1 shows examples of the Type A error and the Type B error. In the remaining part of this chapter, we use these definitions of the Type A and the Type B errors.

The Type A errors are usually involved in genomic and transcriptomic screens since we practically use sequence-based aligners due to the high computational cost for the construction of structural alignment data. In accordance with this convention, the original papers of RNAz and Profile stem kernel tested their methods only on sequence-based alignment datasets (Washietl *et al.*, 2005; Sato *et al.*, 2008). On the other hand, some studies have since then attempted to detect ncRNAs from structural alignment data obtained by realigning sequence-based alignments (Torarinsson *et al.*, 2006, 2008). Following these efforts, the recent update of RNAz has reported the results that its accuracy slightly improved when using structural alignment data as input (Gruber *et al.*, 2010). However, the results were only on the dataset with various ncRNA families mixed, and the improvement for each particular family was not shown. For Profile stem kernel, similar experiments on the Type A errors have not been presented.

The amount of the type B errors has been intensively studied using the 17-way

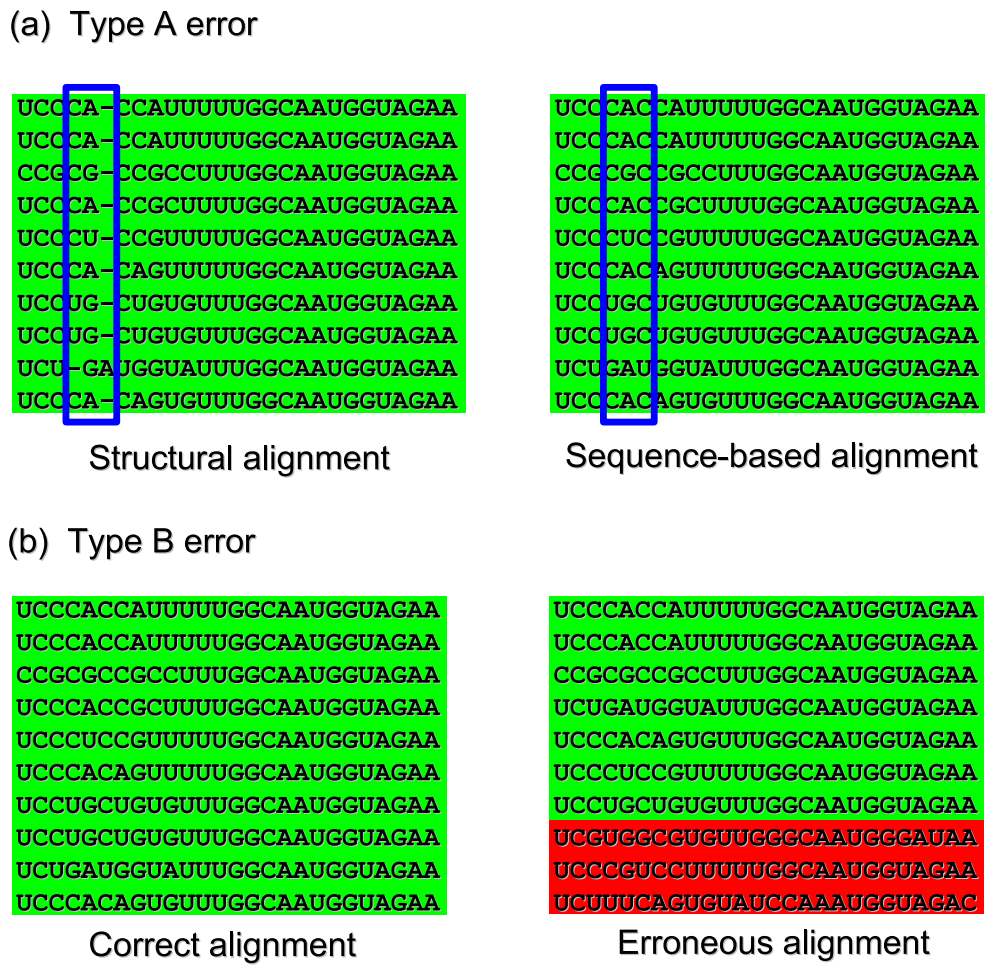


Figure 2.1 Examples of the Type A error and the Type B error in alignment data. Sequences colored with green are miRNA precursors, while those colored with red are unrelated sequences which are not ncRNAs. (a) Example of the Type A error. The left alignment is produced by RAF (Do *et al.*, 2008), which is one of the most accurate tool for structural alignment. The right alignment is produced by CLUSTALW (Thompson *et al.*, 1994) without considering secondary structure. The discrepancy between the two alignments is marked by blue boxes. We call this discrepancy as the Type A error. (b) Example of the Type B error. The left alignment is produced by CLUSTALW, and thus may contain the Type A error. Nevertheless, the left alignment does not contain the Type B error because all sequences in alignment are actually ncRNAs. The right alignment contains three unrelated sequences which can be considered as the Type B error.

vertebrate alignment in the UCSC genome browser (Kuhn *et al.*, 2009). One study has estimated that 9.7% of the regions include unrelated sequences which are not orthologous to the other sequences in the alignment (Prakash and Tompa, 2007). More strikingly, the estimate in (Wang *et al.*, 2007) says that 16% of the segments aligned to ncRNA genes are wrongly included in the alignments from the viewpoint of their secondary structures. In spite of the great significance of the Type B errors suggested by these studies, there has been so far no systematic evaluation about their influence to the performance of prediction methods.

In this chapter, we describe a new method, called Profile BPLA kernel, which predicts ncRNAs from alignment data in combination with SVMs. Profile BPLA kernel is an extension of *base-pairing profile local alignment* (BPLA) kernel which we previously developed for the prediction from single sequences (Morita *et al.*, 2009). By utilizing the profile information of alignment data, the proposed kernel can achieve better accuracy than the original BPLA kernel. We show that Profile BPLA kernel outperforms the existing prediction methods which also utilize the profile information using the high-quality structural alignment dataset. In addition to these standard benchmark tests, we extensively evaluate the robustness of Profile BPLA kernel against errors in input alignments. For both the Type A and the Type B errors, the effects on the performance of Profile BPLA kernel are surprisingly small. Especially for the Type B errors, we demonstrate that Profile BPLA kernel is more robust compared to the existing prediction methods.

2.2 Methods

In this section, we propose an accurate and robust method for the prediction of ncRNAs from alignment data. The proposed method, named Profile BPLA kernel, is an extension of BPLA kernel which we previously developed for the prediction from single sequences (Morita *et al.*, 2009). Hence, we first review the original algorithm of BPLA kernel, and then extend the method to alignment data.

The whole schemes of the original BPLA kernel and Profile BPLA kernel are summarized in Figure 2.2.

(a) Original BPLA kernel

```
INPUT:
training data (set of RNA sequences)
test data (set of RNA sequences)

OUTPUT:
SVM class probability for each of test data


(1) Training

for each sequence x in training data
  compute a base-pairing probability matrix P_x;
  for each position i in x
    compute a base-pairing profile {P_x^L(i), P_x^R(i), P_x^U(i)};
  end for
end for

for each sequence x in training data
  for each sequence y in training data
    compute a value of BPLA kernel K^train(x,y);
  end for
end for

Train a SVM classifier using K^train;


(2) Test

for each sequence x in test data
  compute a base-pairing probability matrix P_x;
  for each position i in x
    compute a base-pairing profile {P_x^L(i), P_x^R(i), P_x^U(i)};
  end for
end for

for each sequence x in test data
  for each sequence y in training data
    compute a value of BPLA kernel K^test(x,y);
  end for
end for

for each sequence x in test data
 compute a SVM class probability for x \
   using K^test and the trained classifier;
end for
```

(b) Profile BPLA kernel

```
INPUT:
training data (set of RNA alignments)
test data (set of RNA alignments)

OUTPUT:
SVM class probability for each of test data


(1) Training

for each alignment X in training data
  P_X = COMPUTE_AVERAGED_BP_MATRIX(X);
  for each column i in X
    compute a base-pairing profile {P_X^L(i), P_X^R(i), P_X^U(i)};
  end for
end for

for each alignment X in training data
  for each alignment Y in training data
    compute a value of Profile BPLA kernel K^train(X,Y);
  end for
end for

Train a SVM classifier using K^train;


(2) Test

for each alignment X in test data
  P_X = COMPUTE_AVERAGED_BP_MATRIX(X);
  for each column i in X
    compute a base-pairing profile {P_X^L(i), P_X^R(i), P_X^U(i)};
  end for
end for

for each alignment X in test data
  for each alignment Y in training data
    compute a value of Profile BPLA kernel K^test(X,Y);
  end for
end for

for each alignment X in test data
 compute a SVM class probability for X \
   using K^test and the trained classifier;
end for

function COMPUTE_AVERAGED_BP_MATRIX(X)
  for each sequence X^k in X
    compute a base-pairing probability matrix P_xk;
  end for
  return the averaged matrix of P_xk;
end function
```

Figure 2.2 Overview of the original BPLA kernel and Profile BPLA kernel.

2.2.1 Notations

For an RNA sequence $\mathbf{x}$, we denote its length by $|\mathbf{x}|$, and the nucleotide at the i-th position by x_i. For a pair of sequences, $\mathbf{x}$ and $\mathbf{y}$, we denote the set of all possible local alignments in the Smith-Waterman (SW) algorithm (Smith and Waterman, 1981) by $\Pi_{\mathbf{xy}}$, and one particular local alignment in $\Pi_{\mathbf{xy}}$ by $\pi_{\mathbf{xy}}$. We denote the alignment score of $\pi_{\mathbf{xy}}$ by $\text{Score}(\pi_{\mathbf{xy}})$, which is calculated based on a scoring function $S_{\mathbf{xy}}(i,j)$ for matching the i-th position in $\mathbf{x}$ and the j-th position in $\mathbf{y}$. We design $S_{\mathbf{xy}}(i,j)$ using a nucleotide substitution matrix $s(x_i, y_j)$ as its component.

For each sequence $\mathbf{x}$, we denote the set of all possible secondary structures by $\Theta_{\mathbf{x}}$, and one particular secondary structure in $\Theta_{\mathbf{x}}$ by $\theta_{\mathbf{x}}$. We represent a secondary structure by $\theta_{\mathbf{x}} = \{\theta_{\mathbf{x}}(i,j)\}_{i<j}$, where a binary variable $\theta_{\mathbf{x}}(i,j)$ is equal to one only when the i-th position and the j-th position in $\mathbf{x}$ form a base pair. In addition, for each position i in $\mathbf{x}$, we define three kinds of binary variable: $L_{\mathbf{x}}(i) = \sum_{j:j>i} \theta_{\mathbf{x}}(i,j)$ is equal to one only when a pair is formed with one of the downstream positions; $R_{\mathbf{x}}(i) = \sum_{j:j<i} \theta_{\mathbf{x}}(j,i)$ is equal to one only when a pair is formed with one of the upstream positions; and $U_{\mathbf{x}}(i) = 1 - L_{\mathbf{x}}(i) - R_{\mathbf{x}}(i)$ is equal to one only when the position is unpaired. These binary variables are converted to the corresponding probabilities by taking the expectation over $\Theta_{\mathbf{x}}$. For $\theta_{\mathbf{x}}(i,j)$, we obtain a base-pairing probability matrix, which consists of the probabilities $P_{\mathbf{x}}(i,j)$ that the i-th and the j-th positions form a base pair:

$$P_{\mathbf{x}}(i,j) = \sum_{\theta_{\mathbf{x}} \in \Theta_{\mathbf{x}}} \theta_{\mathbf{x}}(i,j) P(\theta_{\mathbf{x}}|\mathbf{x}),$$

where the probability distribution $P(\theta_{\mathbf{x}}|\mathbf{x})$ is computed with the McCaskill algorithm (McCaskill, 1990) based on thermodynamic energy models. For $\{L_{\mathbf{x}}(i), R_{\mathbf{x}}(i), U_{\mathbf{x}}(i)\}$, we obtain a *base-pairing profile* (Bonhoeffer *et al.*, 1993), which consists of the probabilities $\{P^L_{\mathbf{x}}(i), P^R_{\mathbf{x}}(i), P^U_{\mathbf{x}}(i)\}$ that the i-th position is paired with one of the downstream/upstream positions, or unpaired, respectively:

$$\begin{aligned}
P^L_{\mathbf{x}}(i) &= \sum_{\theta_{\mathbf{x}} \in \Theta_{\mathbf{x}}} L_{\mathbf{x}}(i) P(\theta_{\mathbf{x}}|\mathbf{x}) = \sum_{\theta_{\mathbf{x}} \in \Theta_{\mathbf{x}}} \sum_{j:j>i} \theta_{\mathbf{x}}(i,j) P(\theta_{\mathbf{x}}|\mathbf{x}) = \sum_{j:j>i} P_{\mathbf{x}}(i,j), \\
P^R_{\mathbf{x}}(i) &= \sum_{\theta_{\mathbf{x}} \in \Theta_{\mathbf{x}}} R_{\mathbf{x}}(i) P(\theta_{\mathbf{x}}|\mathbf{x}) = \sum_{\theta_{\mathbf{x}} \in \Theta_{\mathbf{x}}} \sum_{j:j<i} \theta_{\mathbf{x}}(j,i) P(\theta_{\mathbf{x}}|\mathbf{x}) = \sum_{j:j<i} P_{\mathbf{x}}(j,i), \\
P^U_{\mathbf{x}}(i) &= \sum_{\theta_{\mathbf{x}} \in \Theta_{\mathbf{x}}} U_{\mathbf{x}}(i) P(\theta_{\mathbf{x}}|\mathbf{x}) = 1 - P^L_{\mathbf{x}}(i) - P^R_{\mathbf{x}}(i).
\end{aligned}$$

For a multiple alignment $\mathbf{X}$, we denote the i-th column by X_i, and the k-th sequence by $\mathbf{X}^k$. The nucleotide at the i-th position in $\mathbf{X}^k$ is denoted by X^k_i, which can be a gap character.

2.2.2 Original BPLA kernel for single sequences

A kernel function is a measure of similarity between a pair of objects and can be used as a prediction method in combination with an SVM classifier as long as the Mercer's condition is satisfied (Vapnik, 1998). BPLA kernel calculates the similarity between a pair of RNA sequences using the information of their primary sequences and secondary structures.

The basic idea of BPLA kernel is to perform a pairwise alignment and then to regard the alignment score as the measure of similarity. Instead of relying on one optimal alignment, we accumulate the scores of all possible local alignments in the SW algorithm using *local alignment* (LA) kernel (Saigo *et al.*, 2004). LA kernel between two sequences, $\mathbf{x}$ and $\mathbf{y}$, is defined as follows:

$$K(\mathbf{x}, \mathbf{y}) = \sum_{\pi_{\mathbf{xy}} \in \Pi_{\mathbf{xy}}} e^{\beta \mathrm{Score}(\pi_{\mathbf{xy}})}, \qquad \text{(Eq. 2.1)}$$

where $\beta \geq 0$ is a parameter. In practice, kernel values are normalized to range from 0 to 1:

$$K_n(\mathbf{x}, \mathbf{y}) = \frac{K(\mathbf{x}, \mathbf{y})}{\sqrt{K(\mathbf{x}, \mathbf{x}) K(\mathbf{y}, \mathbf{y})}}. \qquad \text{(Eq. 2.2)}$$

Figure 2.3 shows the state transition diagram of pairwise local alignment. Given a scoring function $S_{\mathbf{xy}}(i, j)$ for the alignment score $\mathrm{Score}(\pi_{\mathbf{xy}})$, LA kernel (Eq. 2.1) can be computed by the following algorithm:

Initialization:
for $i \in \{0, \ldots, |\mathbf{x}|\}$ and $j \in \{0, \ldots, |\mathbf{y}|\}$ **do**
$M(i,0) = I_X(i,0) = I_Y(i,0) = T_X(i,0) = T_Y(i,0) = 0$
$M(0,j) = I_X(0,j) = I_Y(0,j) = T_X(0,j) = T_Y(0,j) = 0$
end for
Iteration:
for $i \in \{1, \ldots, |\mathbf{x}|\}$ and $j \in \{1, \ldots, |\mathbf{y}|\}$ **do**
$M(i,j) = e^{\beta S_{\mathbf{xy}}(i,j)}(1 + I_X(i-1,j-1) + I_Y(i-1,j-1) + M(i-1,j-1))$
$I_X(i,j) = e^{\beta g} M(i-1,j) + e^{\beta d} I_X(i-1,j)$
$I_Y(i,j) = e^{\beta g}(M(i,j-1) + I_X(i,j-1)) + e^{\beta d} I_Y(i,j-1)$
$T_X(i,j) = M(i-1,j) + T_X(i-1,j)$
$T_Y(i,j) = M(i,j-1) + T_X(i,j-1) + T_Y(i,j-1)$
end for
Termination:
$K(\mathbf{x}, \mathbf{y}) = 1 + T_X(|\mathbf{x}|, |\mathbf{y}|) + T_Y(|\mathbf{x}|, |\mathbf{y}|) + M(|\mathbf{x}|, |\mathbf{y}|)$

where the parameters g and d are penalties for gap opening and gap extension, respectively.

To incorporate secondary structure information into the match score $S_{\mathbf{xy}}(i, j)$, we employ the scoring function used in STRAL (Dalli *et al.*, 2006). For each sequence

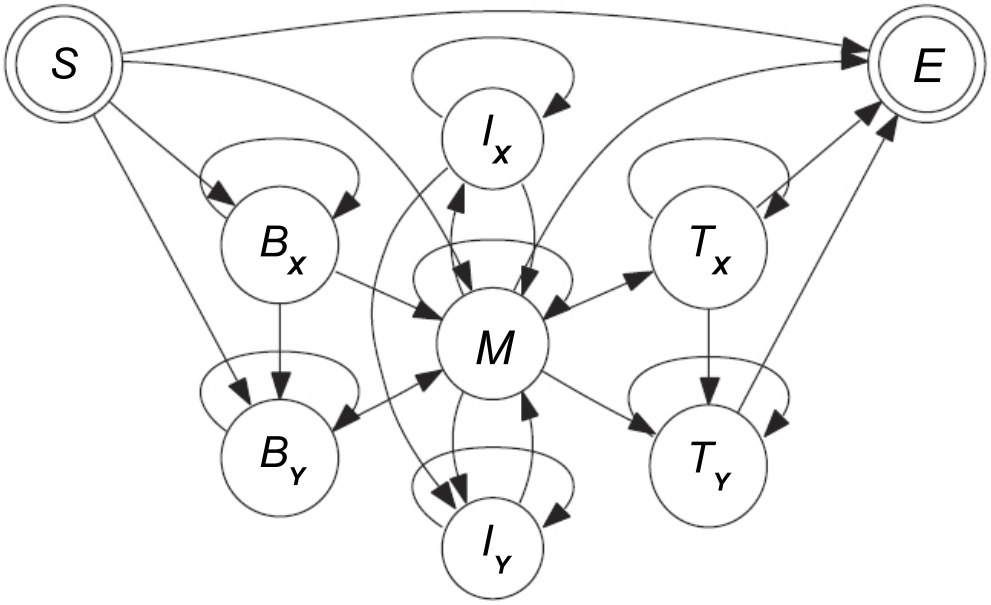


Figure 2.3 The state transition diagram of pairwise local alignment. S is the initial state, B_X and B_Y are the unaligned states before the alignment, M is the match state, I_X and I_Y are the gap states, T_X and T_Y are the unaligned states after the alignment, and E is the final state.

$\mathbf{x}$, we first compute a base-pairing probability matrix $P_{\mathbf{x}}(i,j)$ using the Vienna RNA package (Hofacker, 2003) which is an implementation of the McCaskill algorithm. Subsequently, for each position i in $\mathbf{x}$, we summarize the base-pairing probabilities into the base-pairing profile $\{P^L_{\mathbf{x}}(i), P^R_{\mathbf{x}}(i), P^U_{\mathbf{x}}(i)\}$. We define the scoring function $S_{\mathbf{xy}}(i,j)$ using the base-pairing profiles as follows:

$$\begin{aligned} S_{\mathbf{xy}}(i,j) &= \alpha S_{\text{struct}} + S_{\text{seq}} \\ &= \alpha \left(\sqrt{P^L_{\mathbf{x}}(i) P^L_{\mathbf{y}}(j)} + \sqrt{P^R_{\mathbf{x}}(i) P^R_{\mathbf{y}}(j)} \right) \\ &\quad + s(x_i, y_j) \sqrt{P^U_{\mathbf{x}}(i) P^U_{\mathbf{y}}(j)}, \end{aligned} \qquad \text{(Eq. 2.3)}$$

where $\alpha \geq 0$ is a weight parameter for structural information, and a nucleotide substitution score $s(x_i, y_j)$ captures the similarity of primary sequences. We use the RIBOSUM 85–60 substitution matrix (Klein and Eddy, 2003) as $s(x_i, y_j)$ with the minor modification that its smallest eigenvalue is subtracted from each of its diagonal elements in order to satisfy the Mercer's condition.

Combining LA kernel (Eq. 2.2) with the scoring function (Eq. 2.3), we call this method *base-pairing profile local alignment* (BPLA) kernel.

2.2.3 Profile BPLA kernel for alignment data

Now we extend BPLA kernel to the prediction from alignment data. Profile BPLA kernel for alignment data need to define the similarity between a pair of alignment data instead of a pair of single sequences. More specifically, the new algorithm needs to perform pairwise alignments between two alignment data, and calculate their alignment scores. This problem reduces to the definition of a scoring function corresponding to (Eq. 2.3) for two alignment columns instead of two sequence positions. Both S_{struct} and S_{seq} in (Eq. 2.3) should be extended to take into account the profile information contained in the alignment columns.

In order to define the structural similarity S_{struct} between two alignment columns, we need a base-pairing profile for each alignment column. This can be calculated if we define a base-pairing probability matrix for a multiple alignment. As shown in (Kiryu *et al.*, 2007; Hamada *et al.*, 2009), the consensus secondary structures of aligned sequences are accurately modeled by averaging the individual base-pairing probability matrices. Thus, we define a base-pairing probability matrix for a multiple alignment $\mathbf{X}$ as follows:

$$P_{\mathbf{X}}(i,j) = \frac{1}{N(\mathbf{X})} \sum_{k=1}^{N(\mathbf{X})} P'_{\mathbf{X}^k}(i,j),$$

$$P'_{\mathbf{X}^k}(i,j) = \begin{cases} P_{\mathbf{X}^{k'}}(r(i), r(j)) & (\text{either of } X_i^k \text{ or } X_j^k \text{ is not a gap}) \\ 0 & (\text{otherwise}), \end{cases}$$

where $\mathbf{X}^{k'}$ is the original sequence of $\mathbf{X}^k$ without gaps, $r(i)$ is the index in $\mathbf{X}^{k'}$ corresponding to the i-th position in $\mathbf{X}^k$, and $N(\mathbf{X})$ is the number of aligned sequences in $\mathbf{X}$.

The sequence similarity S_{seq} can be extended by defining a substitution score $s(\cdot,\cdot)$ between two alignment columns. We use the averaged score of all possible substitutions between two columns, X_i and Y_j:

$$s(X_i, Y_j) = \frac{1}{N(\mathbf{X})N(\mathbf{Y})} \sum_{k=1}^{N(\mathbf{X})} \sum_{l=1}^{N(\mathbf{Y})} s'(X_i^k, Y_j^l),$$

$$s'(X_i^k, Y_j^l) = \begin{cases} s(X_i^k, Y_j^l) & (\text{either of } X_i^k \text{ or } Y_j^l \text{ is not a gap}) \\ 0 & (\text{otherwise}). \end{cases}$$

This is equivalent to the sum-of-pairs score, which is widely used in the problem of group-to-group alignment for primary sequences.

Table 2.1 Summary of the combined Rfam families.

Family	NF	N	NS
C/D snoRNA	340	272	5
H/ACA snoRNA	133	119	5
miRNA precursor	401	431	5
Riboswitch	10	85	3
tRNA	1	83	3

Family: name of the larger category used in the performance evaluation. NF: number of smaller families in the Rfam database which were combined. N: number of positive samples. NS: average number of aligned sequences per sample.

2.3 Results and discussion

In this section, we examine the accuracy of Profile BPLA kernel in comparison to the state-of-the-art prediction methods based on SVMs. Furthermore, we present a systematic evaluation about the robustness of Profile BPLA kernel against the Type A and the Type B errors in input alignments. See Background for the definitions of the Type A and the Type B errors.

2.3.1 Dataset and experimental system

We created a dataset which includes 990 positive samples from five ncRNA families: C/D snoRNAs, H/ACA snoRNAs, miRNA precursors, riboswitches, and tRNAs. These families were collected by combining 885 smaller families in the Rfam database (Gardner *et al.*, 2009) into larger categories (Table 2.1). Each positive sample is an alignment of ncRNAs, and is separated by a sequence identity of less than 60% from the other alignment data (see Experimental details). For the construction of input alignments, we produced two versions of the dataset: the high-quality structural alignments by RAF (Do *et al.*, 2008), and the sequence-based alignments by CLUSTALW (Thompson *et al.*, 1994). We generated negative samples which have the same dinucleotide contents as the positives using the randomization by SISSIz (Gesell and Washietl, 2008).

The accuracy of the prediction methods was assessed by the area under the receiver operating characteristic (ROC) curve, *i.e.*, the AUC. The ROC curve plots the true positive rate $TP/(TP+FN)$ versus false positive rate $FP/(TN+FP)$ for different decision thresholds of a SVM classifier, where TP is the number of correctly predicted positives, FP is the number of incorrectly predicted positives, TN is the number of correctly predicted negatives, and FN is the number of incorrectly predicted negatives. We used four-fold cross-validation with the following modifications. The SVM classifier was trained with the same number of negative samples as the positives, and tested on a data partition which includes eight times as many negative samples as the positives. This problem setting is analogous to genomic and transcriptomic

Table 2.2 Accuracy improvement by the profile information.

	AUC (stdev)	
Family	Original BPLA kernel	Profile BPLA kernel
C/D snoRNA	0.91 (0.02)	0.95 (0.02)
H/ACA snoRNA	0.93 (0.03)	0.97 (0.02)
miRNA precursor	0.96 (0.01)	0.97 (0.01)
Riboswitch	0.86 (0.04)	0.92 (0.05)
tRNA	0.98 (0.02)	1.00 (0.00)
Average	0.93 (0.02)	0.96 (0.02)

Family: name of the target ncRNA family. AUC: area under the ROC curve. Profile BPLA kernel, which utilizes the profile information of alignment data, is compared to the original BPLA kernel for single sequences.

screens, where the vast majority of the search space does not contain functional ncRNA genes. Moreover, the four-fold cross validation is repeated four times with different splits of the dataset (16 trials in total). The parameters α, β, g, and d in Profile BPLA kernel were adapted to the training data using the gradient-based optimization developed for the original BPLA kernel (Sato *et al.*, 2009). Note that we did not used the test data for the parameter optimization to avoid overfitting.

2.3.2 Accuracy improvement by the profile information

We first examined whether the proposed kernel could achieve better accuracy than the original BPLA kernel by utilizing the profile information of alignment data. For this purpose, the dataset of single sequences was created from the alignment dataset described above. For positive samples, we randomly chose one sequence from each alignment of ncRNAs. We generated negative samples which have the same dinucleotide contents as the positives by the standard shuffling procedure (Altschul and Erickson, 1985). Then, the proposed kernel and the original BPLA kernel were compared using the high-quality structural alignment dataset and the corresponding single sequence dataset, respectively.

Table 2.2 presents the experimental results. As expected, the proposed kernel achieved the better AUC than the original BPLA kernel for the all ncRNA families. These results suggest that the profile information contained in alignment data is useful to improve the prediction of ncRNAs.

2.3.3 Accuracy on the high-quality structural alignment dataset

Next, we compared Profile BPLA kernel with the existing prediction methods which also utilize the profile information. In the ideal condition, the profile information should be extracted from high-quality alignment data such that all sequences are actually ncRNAs and aligned taking into account their secondary structures. Therefore, we tested the accuracy of each prediction method using the high-quality structural alignment dataset constructed by RAF. The competitors were RNAz

Table 2.3 Accuracy on the high-quality structural alignment dataset.

	AUC (stdev)			
Family	Profile BPLA kernel	Profile LA kernel	Profile stem kernel	RNAz
C/D snoRNA	0.95 (0.02)	0.79 (0.04)	0.80 (0.02)	0.78 (0.03)
H/ACA snoRNA	0.97 (0.02)	0.65 (0.20)	0.89 (0.04)	0.95 (0.03)
miRNA precursor	0.97 (0.01)	0.69 (0.02)	0.92 (0.01)	0.96 (0.01)
Riboswitch	0.92 (0.05)	0.41 (0.23)	0.77 (0.05)	0.97 (0.02)
tRNA	1.00 (0.00)	0.88 (0.03)	0.95 (0.02)	0.96 (0.02)
Average	0.96 (0.02)	0.69 (0.10)	0.86 (0.03)	0.92 (0.02)

Family: name of the target ncRNA family. AUC: area under the ROC curve. Profile BPLA kernel is compared to the other prediction methods which also utilize the profile information of alignment data: Profile LA kernel, Profile stem kernel, and RNAz.

Table 2.4 Accuracy on the sequence-based alignment dataset.

	AUC (stdev)			
Family	Profile BPLA kernel	Profile LA kernel	Profile stem kernel	RNAz
C/D snoRNA	0.95 (0.01)	0.80 (0.04)	0.80 (0.02)	0.77 (0.02)
H/ACA snoRNA	0.96 (0.02)	0.77 (0.17)	0.87 (0.03)	0.94 (0.03)
miRNA precursor	0.97 (0.01)	0.69 (0.03)	0.92 (0.02)	0.96 (0.01)
Riboswitch	0.92 (0.03)	0.38 (0.19)	0.79 (0.05)	0.94 (0.02)
tRNA	1.00 (0.00)	0.88 (0.03)	0.94 (0.03)	0.95 (0.02)
Average	0.96 (0.02)	0.70 (0.09)	0.86 (0.03)	0.91 (0.02)

Family: name of the target ncRNA family. AUC: area under the ROC curve. Profile BPLA kernel is compared to the other prediction methods which also utilize the profile information of alignment data: Profile LA kernel, Profile stem kernel, and RNAz.

(Washietl *et al.*, 2005; Gruber *et al.*, 2010) and Profile stem kernel (Sato *et al.*, 2008). We also performed the experiment with the profile version of LA kernel, which does not consider secondary structure information, by setting base-pairing profiles $\{L_{\mathbf{x}}(i) = 0, R_{\mathbf{x}}(i) = 0, U_{\mathbf{x}}(i) = 1\}$ in Profile BPLA kernel.

Table 2.3 presents the experimental results. Profile BPLA kernel outperformed the other prediction methods except for riboswitches, and achieved the best AUC on average. The accuracy of Profile LA kernel was severely limited compared to the prediction methods which consider secondary structure information. However, for C/D snoRNAs, Profile LA kernel resulted in the comparable AUC with RNAz and Profile stem kernel. These results suggest that RNAz and Profile stem kernel may fail to incorporate the effective information of secondary structures. Profile BPLA kernel consistently achieved the better AUC than Profile LA kernel, showing its wide applicability.

The superiority of Profile BPLA kernel is inherited from the original BPLA kernel. In our previous paper (Morita *et al.*, 2009), we have proved that the original BPLA kernel outperforms the non-profile versions of Stem kernel and LA kernel. Our results showed the high accuracy of BPLA kernels in the prediction from alignment data as

well as from single sequences. (Note that the non-profile version of RNAz does not exist since the feature values of alignment data used in the method can not be defined for single sequences.)

2.3.4 Robustness against the Type A errors

In addition to the standard benchmark tests, we extensively evaluated the robustness of Profile BPLA kernel against errors in input alignments. To discuss the Type A errors, we performed the experiment using the sequence-based alignment dataset constructed by CLUSTALW instead of the high-quality structural alignment dataset.

By comparing the results in Table 2.4 with those in Table 2.3, we can see the robustness of each prediction method against the Type A errors. Profile BPLA kernel achieved almost the same AUC for the two datasets, showing the comparable robustness to RNAz and Profile stem kernel.

The robustness of Profile BPLA kernel can be attributed to its formulation. Profile BPLA kernel utilizes averaged base-pairing probability matrices to obtain the profile information of secondary structures. Averaged base-pairing probability matrices have been shown to be useful for the robust modeling of consensus secondary structures against the Type A errors (Kiryu *et al.*, 2007). Our results showed the effectiveness of averaging base-pairing probabilities for the robustness in the problem of ncRNA prediction.

Our experiment provided the detailed evaluation of the robustness for each particular ncRNA family. The recent study has reported that the accuracy of RNAz can be slightly improved by the use of structural alignment data (Gruber *et al.*, 2010). However, the experiment in (Gruber *et al.*, 2010) has been performed on the dataset with various families mixed. In our experiment, we found that the Type A errors had different effects on the performance of each prediction method depending on families. This in-depth view of the robustness is especially important when we target a particular family in genomic and transcriptomic screens.

Our results also demonstrated that Profile BPLA kernel outperformed the existing prediction methods in the "realistic" condition considered in the previous studies (Washietl *et al.*, 2005; Gruber *et al.*, 2010; Sato *et al.*, 2008). Profile BPLA kernel achieved the best AUC for the sequence-based alignment dataset with the Type A errors as well as for the high-quality structural alignment dataset. In the following experiments, we further evaluated the robustness of Profile BPLA kernel against the Type B errors which have been neglected in the previous studies.

2.3.5 Robustness against the Type B errors

For the systematic evaluation of the robustness, we prepared a controlled series of alignment data with different degrees of the Type B errors. Input alignments in genomic and transcriptomic screens are typically constructed by sequence-based alignment tools. Hence, alignment data with the Type B errors are expected to be optimal at least under the criteria of sequence-based alignment tools, even though

incorrect from the viewpoint of secondary structures. Based on this assumption, we generated sequences which can be well aligned to a given alignment in terms of primary sequences, but do not conserve its consensus secondary structure (see Experimental details). By introducing these "unrelated" sequences, we simulated the Type B errors in the sequence-based alignment dataset. For each positive sample in the test data, a series of erroneous alignments was prepared by gradually replacing ncRNA sequences with unrelated sequences. We aligned the unrelated sequences with the remaining ncRNA sequences using CLUSTALW. The resulting alignments were then used to make the equal-size datasets for the different fractions of unrelated sequences ranged from 0.0 to 1.0 at intervals of 0.1. An alignment comprising n ncRNA sequences and m unrelated sequences was included in the dataset of the fraction f satisfying $(m-1)/(n+m) < f \leq m/(n+m)$. We trained the SVM classifiers with the original training data in the sequence-based alignment dataset, and tested them on the datasets with the different degrees of the simulated Type B errors. The performance was assessed by the AUC for discriminating the erroneous alignments from the alignments consisting only of unrelated sequences.

The experimental results are shown in Figure 2.4. In this figure, zero in the horizontal axis is equivalent to an ordinary prediction problem in which alignments to be discriminated from negative samples do not contain any unrelated sequences. In this situation, Profile BPLA kernel achieved the best accuracy on average, being consistent with the results in Table 2.4. (The AUC, however, were not exactly the same as those in Table 2.4 since we used the different kind of negative samples in the test data between the two experiments: alignments consisting only of unrelated sequences for Figure 2.4, and dinucleotide-controlled samples for Table 2.4.) As the fraction of unrelated sequences increased, the AUC for RNAz rapidly fell down to the baseline. In contrast, Profile BPLA kernel kept the discrimination at high levels until the alignments were overwhelmed by the Type B errors. A similar tendency was seen for Profile stem kernel, although its AUC were smaller than Profile BPLA kernel. The performance of Profile LA kernel was seriously damaged by the Type B errors since the method does not consider secondary structures of unrelated sequences. These results suggest that Profile BPLA kernel is the only method which can effectively detect ncRNAs in the presence of the Type B errors.

The observed differences in the robustness among the methods are deeply connected with the rationales behind their predictions. RNAz detects ncRNAs by utilizing the SCI which measures the conservation of secondary structures in an alignment. Therefore, the experimental results for RNAz can be interpreted as showing that unrelated sequences cause noise in a conserved secondary structure. Profile BPLA kernel do not measure the conservation of secondary structures. Instead, we directly calculate the similarity of secondary structures between input alignments and training data. Hence, Profile BPLA kernel can detect an alignment containing only a few ncRNA sequences if they are similar enough to the ncRNAs in training data, even though the alignment itself is not structurally conserved. Figure 2.5 illustrates an example of the Type B errors and its influences on the performance of the prediction methods. Although RNAz accepted the native alignment (Figure 2.5a), it rejected

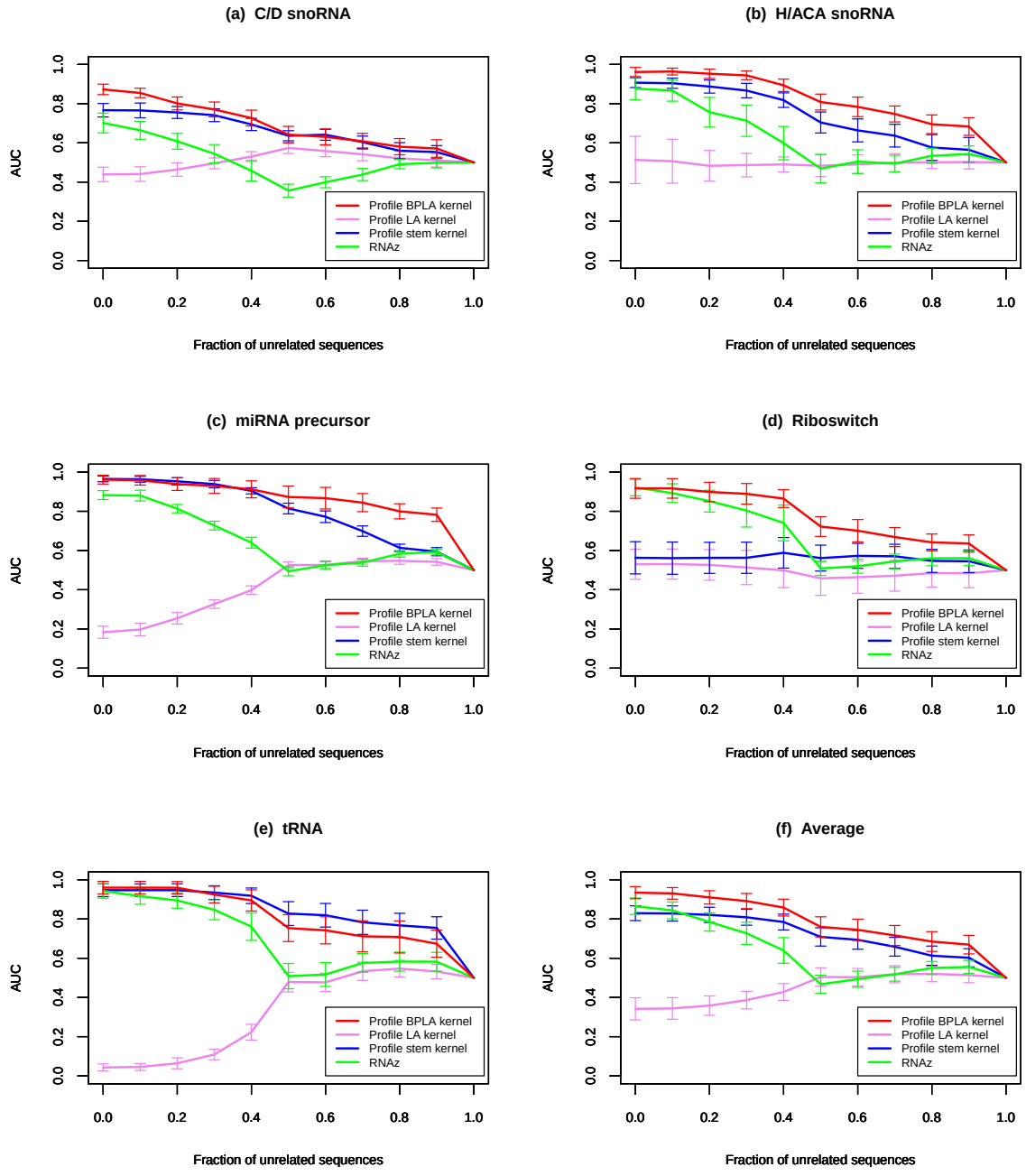


Figure 2.4 Accuracy on the sequence-based alignment dataset with different fractions of unrelated sequences. For each point, the alignments with the different fraction of unrelated sequences were discriminated from the negative samples which consist only of unrelated sequences. Zero in the horizontal axis corresponds to the detection of the alignments which consist only of actual ncRNAs, *i.e.*, an ordinary discrimination problem without the Type B errors.

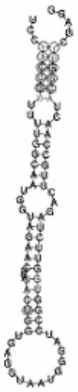
the erroneous alignment (Figure 2.5b) due to the drastic decrease in the SCI value. On the other hand, Profile BPLA kernel kept the SVM class probability moderate for the erroneous alignment, accepting the seven miRNA precursors included in the alignment. Note that the erroneous alignment in Figure 2.5b can be regarded as conserved if we focus only on the sequence identity. This suggests that such alignments can be produced by most alignment tools which do not consider secondary structures. In fact, several studies have suggested that genomic alignments contain significant amounts of the Type B errors (Prakash and Tompa, 2007; Wang *et al.*, 2007). Therefore, the robustness of Profile BPLA kernel is a desirable characteristic for practical applications.

We emphasize that the Type B errors can not be corrected even if we realign the alignments using structural alignment tools as attempted in (Torarinsson *et al.*, 2006, 2008). In contrast to the Type A errors, the Type B errors are caused by the inclusion of unrelated sequences rather than the small shifts of matches and gaps. To make this point clear, we performed the same experiment as in Figure 2.4 and Figure 2.5 using RAF instead of CLUSTALW. For the training data, we used the high-quality structural alignment dataset, and for the test data, we used the erroneous alignment realigned by RAF. As expected, the results in Figure 2.6 and Figure 2.7 were close to those in Figure 2.4 and Figure 2.5, respectively. In Figure 2.6, Profile BPLA kernel outperformed the existing prediction methods for native alignments, and successfully kept the discrimination for alignments with moderate degrees of the Type B errors. Although the erroneous alignment in Figure 2.7b was slightly changed from that in Figure 2.5b, the outputs of the prediction methods were not significantly improved. These results suggest that the problem of the Type B errors is inevitable, and the robustness of Profile BPLA kernel is essential to detect ncRNAs from low-quality alignment data.

(a) 10 miRNAs

Sequence-based alignment with ClustalW

- Mean pairwise identity : 0.85
- Structure conservation index : 0.91
- SVM class probability
  - Profile BPLA kernel : 0.963
  - Profile LA kernel : 0.871
  - Profile stem kernel : 0.716
  - RNAz : 0.935



```
AAQR01061760.1    UCCCACCAUUUUUGGCAAUGGUAGAACUCACACUGGUGAGGUAAUGGGAUCCGGUGGUUCUAGACUUGCCAACUAUGGUGCACGG
CAAA01210023.1    UCCCACCAUUUUUGGCAAUGGUAGAACUCACACCGGUAAGGUAAUGGGACCCGGUGGUUCUAGACUUGCCAACUAUGGUGUAAGU
AAIY01755693.1    CCGCGCCGCCUUUGGCAAUGGUAGAACUCACACUGGUGAGGAAAUGGGAUCCGGUGGUUCUAGACUUGCCAACUACGGUGCCGGG
ABDC01155768.1    UCCCACCGCUUUUUGGCAAUGGUAGAACUCACACUGGUGAGGUCAUGGGAUCCGGUGGUUCUAGACUUGCCAACUCUGGUGCGAGG
AACZ02087971.1    UCCCUCCGUUUUUGGCAAUGGUAGAACUCACACUGGUGAGGUAACAGGAUCCGGUGGUUCUAGACUUGCCAACUAUGGGGCGAGG
AANN01116974.1    UCCCACAGUUUUUGGCAAUGGUAGAACUCACACUGGUGAGGUAAUGGGAUCCGGUGGUUCUAGACUUGCCAACUACGGUGUGAGG
AAPN01078068.1    UCCUGCUGUGUUUGGCAAUGGUAGAACUCACACUGGUGAGGUAAUGGGAUCCGGUGGUUCUAGACUUGCCAACUAUGGCCCGAGG
AAFR03031888.1    UCCUGCUGUGUUUGGCAAUGGUAGAACUCACACUGGUGAGAUAAUGGAAUCCGGUGGUUCUAGACUUGCCAACUACGGCUUGAGA
AL590150.2        UCUGAUGGUAUUUGGCAAUGGUAGAACUCACACUGGUGAGGUAGUCAGAUCCGGUGGUUCUAGACUUGCCAACUACUACCUGAGA
CAAB01003233.1    UCCCACAGUGUUUGGCAAUGGUAGAACUCACACUCCGGUGGGCUAGAAGGAUCCGGUAGUUCUAGAAUUGCCAACAACUGACCGAGA
```

(b) 7 miRNAs + 3 unrelated sequences

Sequence-based alignment with ClustalW

- Mean pairwise identity : 0.76
- Structure conservation index : 0.42
- SVM class probability
  - Profile BPLA kernel : 0.772
  - Profile LA kernel : 0.937
  - Profile stem kernel : 0.206
  - RNAz : 0.191

```
AAQR01061760.1    UCCCACCAUUUUUGGCAAUGGUAGAACUCACACUGGUGAGGUAAUGGGAUCCGGUGGUUCUAGACUUGCCAACUAUGGUGCACGG
CAAA01210023.1    UCCCACCAUUUUUGGCAAUGGUAGAACUCACACCGGUAAGGUAAUGGGACCCGGUGGUUCUAGACUUGCCAACUAUGGUGUAAGU
AAIY01755693.1    CCGCGCCGCCUUUGGCAAUGGUAGAACUCACACUGGUGAGGAAAUGGGAUCCGGUGGUUCUAGACUUGCCAACUACGGUGCCGGG
AL590150.2        UCUGAUGGUAUUUGGCAAUGGUAGAACUCACACUGGUGAGGUAGUCAGAUCCGGUGGUUCUAGACUUGCCAACUACUACCUGAGA
CAAB01003233.1    UCCCACAGUGUUUGGCAAUGGUAGAACUCACACUCCGGUGGGCUAGAAGGAUCCGGUAGUUCUAGAAUUGCCAACAACUGACCGAGA
AACZ02087971.1    UCCCUCCGUUUUUGGCAAUGGUAGAACUCACACUGGUGAGGUAACAGGAUCCGGUGGUUCUAGACUUGCCAACUAUGGGGCGAGG
AAPN01078068.1    UCCUGCUGUGUUUGGCAAUGGUAGAACUCACACUGGUGAGGUAAUGGGAUCCGGUGGUUCUAGACUUGCCAACUAUGGCCCGAGG
UNRELATED_39.17   UCGUGGCGUGUUGGGCAAUGGGAUAACUCACACUGGUGAGGUAGUGGGAUCGGGAGGUUCUAGAAGUCACAACUAUUGUUUGAGA
UNRELATED_36.11   UCCCGUCCUUUUUGGCAAUGGUAGAACGCACACUCGCGAGGUAAUAGGAUCCGAUGAUGAUAGCGUUGAGUACUACGGGCUAAGA
UNRELATED_35.57   UCUUUCAGUGUAUCCAAAUGGUAGACCUCAAUAGGCUGAGGCAUUGGGAUCCGGUGGUUCUAGACGUGCCAACUACGGACUCGGG
```

Figure 2.5 Example of the Type B errors and its influence on the prediction methods. (a) Native alignment consisting only of ncRNAs. An alignment of 10 miRNA precursors is highly conserved in terms of both primary sequences and secondary structures. The consensus secondary structure predicted by RNAalifold (Bernhart *et al.*, 2008) exhibits a well-known hairpin loop. Profile BPLA kernel and the other prediction methods accepted this alignment. (b) Alignment with the Type B errors. Three miRNA precursors in the native alignment were replaced with unrelated sequences, which destroyed the consensus secondary structure. This alignment was rejected by RNAz due to the drastic decrease in the SCI and also missed by Profile stem kernel. Profile LA kernel was completely ruined showing the higher SVM class probability for the erroneous alignment than that for the native one. Profile BPLA kernel was the only method to accept the alignment by the moderate decrease in the SVM class probability from the native one. Note that the mean pairwise identity is still high allowing this alignment to be produced by sequence-based alignment tools.

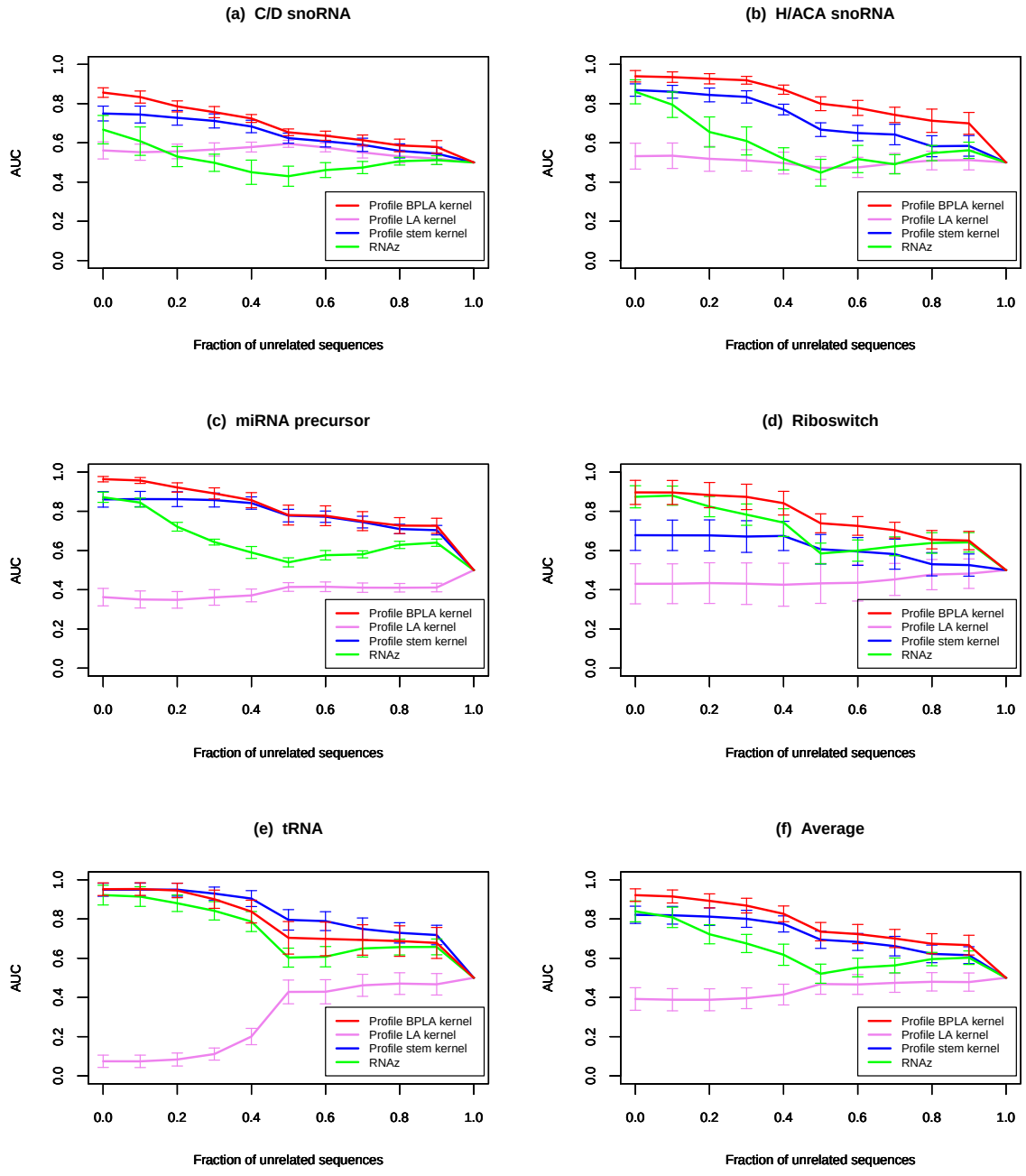
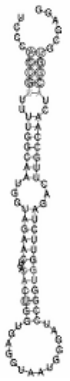


Figure 2.6 Accuracy on the structural alignment dataset with different fractions of unrelated sequences. For each point, the alignments with the different fraction of unrelated sequences were discriminated from the negative samples which consist only of unrelated sequences. Zero in the horizontal axis corresponds to the detection of the alignments which consist only of actual ncRNAs, *i.e.*, an ordinary discrimination problem without the Type B errors.

(a) 10 miRNAs

Structural alignment with RAF

- Mean pairwise identity : 0.85
- Structure conservation index : 0.82
- SVM class probability
  - Profile BPLA kernel : 0.979
  - Profile LA kernel : 0.846
  - Profile stem kernel : 0.643
  - RNAz : 0.946



```
AAQR01061760.1     UCCCA-CCAUUUUUGGCAAUGGUAGAACUCACACUGGUGAGGUAAUGGGAUCCGGUGGUUCUAGACUUGCCAACUAUGGUGCACGG
CAAA01210023.1     UCCCA-CCAUUUUUGGCAAUGGUAGAACUCACACCGGUAAGGUAAUGGGACCCGGUGGUUCUAGACUUGCCAACUAUGGUGUAAGU
AAIY01755693.1     CCGCG-CCGCCUUUGGCAAUGGUAGAACUCACACUGGUGAGGAAAUGGGAUCCGGUGGUUCUAGACUUGCCAACUACGGUGCCGGG
ABDC01155768.1     UCCCA-CCGCUUUUGGCAAUGGUAGAACUCACACUGGUGAGGUCAUGGGAUCCGGUGGUUCUAGACUUGCCAACUCUGGUGCGAGG
AACZ02087971.1     UCCCU-CCGUUUUUGGCAAUGGUAGAACUCACACUGGUGAGGUAACAGGAUCCGGUGGUUCUAGACUUGCCAACUAUGGGGCGAGG
AANN01116974.1     UCCCA-CAGUUUUUGGCAAUGGUAGAACUCACACUGGUGAGGUAAUGGGAUCCGGUGGUUCUAGACUUGCCAACUACGGUGUGAGG
AAPN01078068.1     UCCUG-CUGUGUUUGGCAAUGGUAGAACUCACACUGGUGAGGUAAUGGGAUCCGGUGGUUCUAGACUUGCCAACUAUGGCCCGAGG
AAFR03031888.1     UCCUG-CUGUGUUUGGCAAUGGUAGAACUCACACUGGUGAGAUAAUGGAAUCCGGUGGUUCUAGACUUGCCAACUACGGCUUGAGA
AL590150.2         UCU-GAUGGUAUUUGGCAAUGGUAGAACUCACACUGGUGAGGUAGUCAGAUCCGGUGGUUCUAGACUUGCCAACUACUACCUGAGA
CAAB01003233.1     UCCCA-CAGUGUUUGGCAAUGGUAGAACUCACUCCGGUGGGCUAGAAGGAUCCGGUAGUUCUAGAAUUGCCAACAACUGACCGAGA
```

(b) 7 miRNAs + 3 unrelated sequences

Structural alignment with RAF

- Mean pairwise identity : 0.75
- Structure conservation index : 0.41
- SVM class probability
  - Profile BPLA kernel : 0.803
  - Profile LA kernel : 0.866
  - Profile stem kernel : 0.156
  - RNAz : 0.325

```
AAQR01061760.1     UCC-CACC-AUUUUUG-GCAAUGGUAGAACUCACAC-UGGUGAGGUAAUGGGAUCCGGUGGUUCUAGACUUGCCAACUAUGGUGCACGG
CAAA01210023.1     UCC-CACC-AUUUUUG-GCAAUGGUAGAACUCACAC-CGGUAAGGUAAUGGGACCCGGUGGUUCUAGACUUGCCAACUAUGGUGUAAGU
AAIY01755693.1     CCG-CGCC-GCCUUUG-GCAAUGGUAGAACUCACAC-UGGUGAGGAAAUGGGAUCCGGUGGUUCUAGACUUGCCAACUACGGUGCCGGG
AL590150.2         UCU-GAUG-GUAUUUG-GCAAUGGUAGAACUCACAC-UGGUGAGGUAGUCAGAUCCGGUGGUUCUAGACUUGCCAACUACUACCUGAGA
CAAB01003233.1     UCC-CACA-GUGUUUG-GCAAUGGUAGAACUCACUC-CGGUGGGCUAGAAGGAUCCGGUAGUUCUAGAAUUGCCAACAACUGACCGAGA
AACZ02087971.1     UCC-CUCC-GUUUUUG-GCAAUGGUAGAACUCACAC-UGGUGAGGUAACAGGAUCCGGUGGUUCUAGACUUGCCAACUAUGGGGCGAGG
AAPN01078068.1     UCC-UGCU-GUGUUUG-GCAAUGGUAGAACUCACAC-UGGUGAGGUAAUGGGAUCCGGUGGUUCUAGACUUGCCAACUAUGGCCCGAGG
UNRELATED_39.17    UCGUGGCG--UG-UUGGGCAAUGGGAUAACUCACAC-UGGUGAGGUAGUGGGAUCGGGAGGUUCUAGAAGUCACAACUAUUGUUUGAGA
UNRELATED_36.11    UCCCGUCC-UU-UUUG-GCAAUGGUAGAACGCACACUCG-CGAGGUAAUAGGAUCCGAUGAUGAUAGCGUUGAGUACUACGGGCUAAGA
UNRELATED_35.57    UCU-U-UCAGUGUAUC-CAAAUGGUAGACCUCAAUA-GGCUGAGGCAUUGGGAUCCGGUGGUUCUAGACGUGCCAACUACGGACUCGGG
```

Figure 2.7 Realigning unrelated sequences by structural alignment tools attempting to correct the Type B errors. (a) Native alignment consisting only of ncRNAs. (b) Alignment with the Type B errors. In contrast to the type A errors, the Type B errors cannot be corrected even if we realign the alignments using structural alignment tools. Profile BPLA kernel was still the only method to accept the seven miRNA precursors in the alignment with the Type B errors.

2.4 Experimental details

2.4.1 Combining related Rfam families

We created the datasets for the benchmark tests using the Rfam database (Gardner *et al.*, 2009) version 9.1. To make the tests more challenging, we combined related Rfam families into larger categories as shown in Table 2.1. For example, the C/D snoRNA family in Table 2.1 was established by combining the 340 Rfam families which have the string "snoRNA; CD-box;" in the description track. The seed alignments for these families were then split into single sequences. We performed a complete linkage clustering using their sequence identity as the similarity function. Clusters were determined using the similarity threshold of 60%, and we obtained one alignment from each cluster consisting of multiple sequences.

2.4.2 Generating unrelated sequences

We generated unrelated sequences for simulating the Type B errors in alignment data. For each larger category in Table 2.1, we took the seed alignments of the corresponding smaller Rfam families. For each seed alignment, we constructed a profile hidden Markov model (profile HMM) using HMMER (Eddy, 1998), and a covariance model (CM) using INFERNAL (Nawrocki *et al.*, 2009). Profile HMMs and CMs are grammar models to generate sequences which can be well aligned to given alignments, and to calculate scores for aligning generated sequences to the original alignments. Profile HMMs do not consider the constraints of consensus secondary structures in alignments, whereas CMs do. We generated 100000 sequences from the profile HMM, and calculated the scores for aligning these sequences using the profile HMM and the CM. We needed sequences which can be well aligned to a given alignment, but do not conserve its consensus secondary structure. Therefore, we chose the top 100 sequences whose score difference between the profile HMM and the CM was large, and used them as the pool of unrelated sequences.

2.4.3 Software versions and options

We used the most recent version of each software, and if not specified, executed it with the default options. We used RNAz (Washietl *et al.*, 2005; Gruber *et al.*, 2010) version 2.0 and Profile stem kernel (Sato *et al.*, 2008) version 216c. For the computation of base-pairing probability matrices, we used the Vienna RNA package (Hofacker, 2003) version 1.8.4. To construct the sequence-based and the structural alignment datasets, we used CLUSTALW (Thompson *et al.*, 1994) version 1.83 and RAF (Do *et al.*, 2008) version 1.00, respectively. To generate the negative samples, we used SISSIZ version 0.1 with the option "−−simulate −−tstv −−precision 0.05 −−rna" recommended in the original paper (Gesell and Washietl, 2008). For the prediction of the consensus secondary structures shown in Figure 2.5 and Figure 2.7,

we used RNAalifold (Bernhart *et al.*, 2008) included in the Vienna RNA package version 1.8.4. To simulate the unrelated sequences for the Type B errors, we used the HMMER package (Eddy, 1998) version 2.3.2 and the INFERNAL package (Nawrocki *et al.*, 2009) version 1.0. For the individual programs in the HMMER and the INFERNAL packages, we used the following commands: "hmmbuild −g", "hmmsearch −E 100000", and "cmsearch −g −T −10000 −−toponly −−no−qdb −−fil−no−hmm −−fil−no−qdb". Basically, these options were set because we needed global alignments rather than local alignments for the evaluation of the Type B errors, and wanted to calculate the exact scores for profile HMMs and CMs without several heuristics implemented in the programs.

2.4.4 Availability

Our implementation of Profile BPLA kernel (including the original BPLA kernel for single sequences) is freely available at http://bpla-kernel.dna.bio.keio.ac.jp/ under the GNU general public license. It takes RNA sequences or multiple alignments, and calculates a kernel matrix, which can be used as an input for a popular SVM tool called LIBSVM (Fan *et al.*, 2005). Furthermore, our software is capable of parallel processing using the message passing interface (MPI) (Pacheco, 1996).

2.5 Conclusion

We have described a new method for the prediction of ncRNAs from alignment data. Our method, named Profile BPLA kernel, is an extension of BPLA kernel which was originally developed for the prediction from single sequences (Morita *et al.*, 2009). By utilizing the profile information of alignment data, the proposed kernel can achieve better accuracy than the original BPLA kernel. Furthermore, Profile BPLA kernel outperforms the state-of-the-art prediction methods (Washietl *et al.*, 2005; Gruber *et al.*, 2010; Sato *et al.*, 2008) which also utilize the profile information.

The evaluation of the robustness against errors in input alignments is a crucial step for the development of practical prediction methods. Even with prediction methods showing excellent accuracy for well-curated alignment datasets, the same performance typically cannot be expected in the practical situations which involve significant amounts of alignment errors. Previous studies did not fully address this issue. Through the present study, we extensively evaluated the effectiveness of Profile BPLA kernel under the realistic conditions in which the quality of input alignments is not necessarily high. We considered the two different types of error in alignment data: first, that all sequences in an alignment are actually ncRNAs but are aligned ignoring their secondary structures (Type A); second, that an alignment contains unrelated sequences which are not ncRNAs but still aligned (Type B). Our experiments presented the more detailed evaluation for the Type A errors than the previous study (Gruber *et al.*, 2010), and the first systematic evaluation for the Type B errors. For the Type A errors, Profile BPLA kernel has the comparable robustness to the existing prediction methods. For the Type B errors, Profile BPLA kernel achieves the higher level of robustness than the existing prediction methods.

We conclude that Profile BPLA kernel provides a promising way for identifying ncRNAs genes from alignment data.

Chapter 3

Fast and accurate clustering of noncoding RNAs using ensembles of sequence alignments and secondary structures

In this chapter, we propose a method that finds candidates of novel noncoding RNA (ncRNA) families from a set of unannotated RNAs (Saito *et al.*, 2011). This problem can be considered as an application of similarity search where clustering detects subsets of RNAs which are similar to each other.

Several hierarchical clustering methods have been developed using similarity measures based on the scores of structural alignment. However, the high computational cost of exact structural alignment requires these methods to employ approximate algorithms. Such heuristics degrade the quality of clustering results, especially when the similarity among family members is not detectable at the primary sequence level.

We describe a new similarity measure for the hierarchical clustering of ncRNAs. The idea is that the reliability of approximate algorithms can be improved by utilizing the information of suboptimal solutions in their dynamic programming (DP) frameworks. We approximate structural alignment in a more simplified manner than the existing methods. Instead, our method utilizes *all possible* sequence alignments and *all possible* secondary structures, whereas the existing methods only use *one optimal* sequence alignment and *one optimal* secondary structure. We demonstrate that this strategy can achieve the best balance between the computational cost and the quality of the clustering. In particular, our method can keep its high performance even when the sequence identity of family members is less than 60%.

3.1 Background

Recently, high-throughput transcriptome sequencing has uncovered tens of thousands of ncRNAs that lack significant homology to known families (Guttman *et al.*, 2010; Rederstorff *et al.*, 2010). Thus, evaluating homology *among* these unannotated transcripts, that is, *clustering* has become an important task to identify novel ncRNA families (Shi *et al.*, 2009; Weinberg *et al.*, 2009).

Accurate clustering of ncRNAs needs a reliable similarity measure that takes into account primary sequences and secondary structures. Given a pair of sequences without known structures, the Sankoff algorithm (Sankoff, 1985) simultaneously predicts their sequence alignment and consensus secondary structure (*i.e.*, structural alignment); thus, the obtained alignment score can be a suitable choice for a similarity measure. However, the original Sankoff algorithm is too time-consuming to deal with

an all-against-all comparison of many sequences required in clustering procedures. To address this problem, similarity measures based on the approximation of the Sankoff algorithm have been proposed, and shown to be applicable to hierarchical clustering (Will *et al.*, 2007; Torarinsson *et al.*, 2007; Sato *et al.*, 2008). Each method has its own heuristics to reduce the huge DP matrix used in the Sankoff algorithm. Will *et al.* (2007) have developed LocARNA that precludes unsure secondary structures including low-probability base pairs. Torarinsson *et al.* (2007) have developed FOLDALIGN**M** based on the FOLDALIGN program (Havgaard *et al.*, 2007) that dynamically excludes low-scoring sequence alignments by means of length-dependent thresholds. Sato *et al.* (2008) have developed Stem kernel that employs heuristics similar to LocARNA, but further precludes secondary structures including any bifurcation.

Although the approximate Sankoff-style algorithms have enabled similarity measures based on structural alignment, the quality of clustering results has not been so high. In the previous studies (Will *et al.*, 2007; Weinberg *et al.*, 2009, 2010), resultant clusters in a hierarchical tree were quite unclear, requiring additional verification or manual inspection. This was partly because of the diversity within one ncRNA family. Most ncRNA families have only less than 60% identity at the primary sequence level (Gardner *et al.*, 2009), and cannot be correctly aligned without taking into account secondary structures (Wilm *et al.*, 2006). The approximate Sankoff-style algorithms seemed to be degraded by discarding the secondary structures in the excluded portion of the DP matrix.

To improve the reliability of the approximate Sankoff-style algorithms, we focus on the information of suboptimal structural alignments. Among the existing methods, LocARNA and FOLDALIGN calculate the similarity based on the score of *one optimal* structural alignment. This means that these methods ignore the scores of suboptimal structural alignments, and only use *one optimal* sequence alignment and *one optimal* secondary structure. In contrast, Stem kernel sums up the scores of structural alignments allowed in the approximate Sankoff-style algorithm, incorporating a *subset of* sequence alignments and a *subset of* secondary structures. As a consequence of this strategy, Stem kernel gives comparable clustering results to LocARNA, while employing the more reduced DP matrix. These observations suggest the possibility that we can design a more reliable similarity measure by utilizing *all possible* sequence alignments and *all possible* secondary structures. This is not trivial because if we naively try to incorporate all possible structural alignments, it will require the full-size DP matrix used in the original Sankoff algorithm with the prohibitive computational cost.

In this paper, we describe a new similarity measure for the hierarchical clustering of ncRNAs. We approximate the problem of structural alignment by the two separate problems: the prediction of sequence alignment, and the prediction of secondary structure for each sequence. For this purpose, the Sankoff algorithm for structural alignment is approximated by the combination of the Smith-Waterman (SW) algorithm (Smith and Waterman, 1981) for sequence alignment, and the McCaskill algorithm (McCaskill, 1990) for secondary structures. The approximation

allows to obtain all possible sequence alignments from the SW algorithm, and all possible secondary structures from the McCaskill algorithm, much faster than obtaining all possible structural alignments from the original Sankoff algorithm. We first describe a similarity measure using the scores of all possible sequence alignments between two RNAs. Then, we design a scoring function for these sequence alignments using all possible secondary structures of each of the two RNAs. We start from a scoring function that measures the similarity between two secondary structures using the state of base pairing at each position. The proposed scoring function is defined as an expectation of this scoring function over all possible secondary structures of each of the two RNAs.

We demonstrate that our method can achieve the best balance between the computational cost and the quality of the clustering among the existing methods. In particular, our method can keep its high performance even when the sequence identity of family members is less than 60%.

3.2 Methods

In this section, we propose a new method for measuring the similarity between two RNA sequences without known structures. The proposed method is applied to the hierarchical clustering of ncRNAs with the weighted pair-group method with arithmetic mean (WPGMA) algorithm. Given a set of sequences, we calculate an all-against-all similarity matrix using our method. Then, we derive the distance matrix by one minus the similarity, and obtain the cluster tree by the WPGMA algorithm.

The idea of our similarity measure is to approximate the Sankoff algorithm for structural alignment by the combination of the SW algorithm for sequence alignment, and the McCaskill algorithm for secondary structures. This approximation allows to utilize the ensembles of *all possible* sequence alignments and *all possible* secondary structures separately from each of the two algorithms. First, we describe a similarity measure using the scores of all possible sequence alignments between two RNAs. Next, we design a scoring function for these alignments using all possible secondary structures of each of the two RNAs.

3.2.1 Ensemble of all possible sequence alignments

To measure the similarity between two RNAs, one common approach is to perform pairwise alignment, and to calculate its alignment score. The Sankoff algorithm simultaneously models sequence alignments and secondary structures, and is extremely time-consuming. Therefore, we first approximate the Sankoff algorithm by the SW algorithm that only models sequence alignments apart from secondary structures. Although this is a strong approximation, we attempt to improve the reliability by utilizing *all possible* sequence alignments rather than *one optimal* sequence alignment.

For an RNA sequence $\mathbf{x}$, we denote its length by $|\mathbf{x}|$. For each position $1 \leq i \leq |\mathbf{x}|$ in $\mathbf{x}$, we denote the nucleotide by $x_i \in \{\mathtt{A}, \mathtt{C}, \mathtt{G}, \mathtt{U}\}$.

For two sequences, $\mathbf{x}$ and $\mathbf{y}$, let $\Pi_{\mathbf{xy}}$ be the set of all possible sequence alignments in the SW algorithm. Let $\pi_{\mathbf{xy}}$ denote one particular sequence alignment in $\Pi_{\mathbf{xy}}$.

We calculate the similarity between $\mathbf{x}$ and $\mathbf{y}$ by accumulating the alignment score of $\pi_{\mathbf{xy}}$ over $\Pi_{\mathbf{xy}}$. For this purpose, we employ local alignment (LA) kernel (Saigo *et al.*, 2004) defined as follows:

$$K(\mathbf{x}, \mathbf{y}) = \sum_{\pi_{\mathbf{xy}} \in \Pi_{\mathbf{xy}}} e^{\beta \mathrm{Score}(\pi_{\mathbf{xy}})}, \qquad \text{(Eq. 3.1)}$$

where $\beta \geq 0$ is a parameter, and $\mathrm{Score}(\pi_{\mathbf{xy}})$ is the alignment score of $\pi_{\mathbf{xy}}$ under a given scoring scheme (gap penalties and match scores). In practice, we take the logarithm of LA kernel, and similarity values are normalized to range from 0 to 1:

$$K_n(\mathbf{x}, \mathbf{y}) = \frac{\log K(\mathbf{x}, \mathbf{y})}{\sqrt{\log K(\mathbf{x}, \mathbf{x}) \log K(\mathbf{y}, \mathbf{y})}}. \qquad \text{(Eq. 3.2)}$$

The normalization of the similarity measure in (Eq. 3.2) is different from (Eq. 2.2) in Chapter 2. The logarithm of LA kernel does not satisfy the Mercer's condition that is necessary for using the similarity measure in combination with support vector machines (SVMs). In clustering problems, where SVMs are not applied, we find that (Eq. 3.2) gives slightly better accuracy than (Eq. 2.2).

LA kernel (Eq. 3.1) can be computed by the variant of the SW algorithm as follows:

Initialization:
for $i \in \{0, \ldots, |\mathbf{x}|\}$ and $j \in \{0, \ldots, |\mathbf{y}|\}$ **do**
$M(i,0) = I_X(i,0) = I_Y(i,0) = T_X(i,0) = T_Y(i,0) = 0$
$M(0,j) = I_X(0,j) = I_Y(0,j) = T_X(0,j) = T_Y(0,j) = 0$
end for
Iteration:
for $i \in \{1, \ldots, |\mathbf{x}|\}$ and $j \in \{1, \ldots, |\mathbf{y}|\}$ **do**
$M(i,j) = e^{\beta S_{\mathbf{xy}}(i,j)}(1 + I_X(i-1,j-1) + I_Y(i-1,j-1) + M(i-1,j-1))$
$I_X(i,j) = e^{\beta g} M(i-1,j) + e^{\beta d} I_X(i-1,j)$
$I_Y(i,j) = e^{\beta g}(M(i,j-1) + I_X(i,j-1)) + e^{\beta d} I_Y(i,j-1)$
$T_X(i,j) = M(i-1,j) + T_X(i-1,j)$
$T_Y(i,j) = M(i,j-1) + T_X(i,j-1) + T_Y(i,j-1)$
end for
Termination:
$K(\mathbf{x}, \mathbf{y}) = 1 + T_X(|\mathbf{x}|, |\mathbf{y}|) + T_Y(|\mathbf{x}|, |\mathbf{y}|) + M(|\mathbf{x}|, |\mathbf{y}|)$

where the parameters g and d are the penalties for gap opening and gap extension, respectively, and $S_{\mathbf{xy}}(i,j)$ is a scoring function for matching the i-th position in $\mathbf{x}$ and the j-th position in $\mathbf{y}$. The design of $S_{\mathbf{xy}}(i,j)$ impacts the performance of the resulting similarity measure, and will be described later.

At this point, we note that our method can take into account all possible sequence alignments in $O(|\mathbf{x}||\mathbf{y}|)$ time. If we use the exact Sankoff algorithm instead, it takes prohibitive $O(|\mathbf{x}|^3|\mathbf{y}|^3)$ time, which is not practical. In the case of the approximate Sankoff-style algorithms employed in the existing methods, all possible sequence alignments cannot be incorporated to the reduced DP matrix. Therefore, LA kernel based on the SW algorithm is an efficient way to deal with the ensemble of all possible sequence alignments.

3.2.2 Ensemble of all possible secondary structures

To design a scoring function $S_{\mathbf{xy}}(i,j)$ for LA kernel, we need secondary structures of $\mathbf{x}$ and $\mathbf{y}$. As mentioned above, the Sankoff algorithm models secondary structures simultaneously with sequence alignments which we have already modeled by the SW algorithm. Therefore, we next employ the McCaskill algorithm that only models secondary structures apart from sequence alignments. Although this is an additional approximation, we attempt to improve the reliability by utilizing *all possible* secondary structures rather than *one optimal* secondary structure.

For an RNA sequence x, let $\Theta_{\mathbf{x}}$ be the set of all possible secondary structures. Let $\theta_{\mathbf{x}}$ denote one particular secondary structure in $\Theta_{\mathbf{x}}$. We represent a secondary structure as a set of binary variables $\theta_{\mathbf{x}} = \{\theta_{\mathbf{x}}(i,j)\}_{1\leq i<j\leq|\mathbf{x}|}$, where $\theta_{\mathbf{x}}(i,j) = 1$ means that the i-th position and the j-th position in $\mathbf{x}$ form a base pair. For each position $1 \leq i \leq |\mathbf{x}|$ in $\mathbf{x}$, we represent the state of base-pairing using three kinds of binary variable: $L_{\mathbf{x}}(i) = \sum_{j:j>i} \theta_{\mathbf{x}}(i,j) = 1$ means that a base pair is formed with one of the downstream positions; $R_{\mathbf{x}}(i) = \sum_{j:j<i} \theta_{\mathbf{x}}(j,i) = 1$ means that a base pair is formed with one of the upstream positions; and $U_{\mathbf{x}}(i) = 1 - L_{\mathbf{x}}(i) - R_{\mathbf{x}}(i) = 1$ means that the position is unpaired. Given a fixed pair of secondary structures, $\theta_{\mathbf{x}}$ and $\theta_{\mathbf{y}}$, we can measure the similarity between the i-th position in $\mathbf{x}$ and the j-th position in $\mathbf{y}$ using their state of base pairing:

$$\begin{aligned} W_{\mathbf{xy}}(i,j|\theta_{\mathbf{x}},\theta_{\mathbf{y}}) = \alpha\,(&L_{\mathbf{x}}(i)L_{\mathbf{y}}(j) + R_{\mathbf{x}}(i)R_{\mathbf{y}}(j)) \\ &+ s(x_i,y_j)U_{\mathbf{x}}(i)U_{\mathbf{y}}(j), \end{aligned} \qquad \text{(Eq. 3.3)}$$

where $\alpha \geq 0$ is a weight parameter for structural similarity, and $s(x_i,y_j)$ is a substitution matrix for RNA sequences like the RIBOSUM 85–60 matrix (Klein and Eddy, 2003). This scoring function takes a non-zero value in three different cases: it takes α when both of the two positions form a base pair with one of their downstream positions, respectively; it takes α when both of the two positions form a base pair with one of their upstream positions, respectively; and it takes $s(x_i,y_j)$ when both of the two positions are unpaired.

The McCaskill algorithm defines a probability distribution $P(\theta_{\mathbf{x}}|\mathbf{x})$ over $\Theta_{\mathbf{x}}$. The binary variables $\theta_{\mathbf{x}}(i,j)$ and $\{L_{\mathbf{x}}(i), R_{\mathbf{x}}(i), U_{\mathbf{x}}(i)\}$ are converted to the probabilities by taking the expectation over $\Theta_{\mathbf{x}}$. For $\theta_{\mathbf{x}}(i,j)$, we obtain a base-pairing probability

$P_{\mathbf{x}}(i,j)$ that the i-th and the j-th positions form a base pair:

$$P_{\mathbf{x}}(i,j) = \sum_{\theta_{\mathbf{x}} \in \Theta_{\mathbf{x}}} \theta_{\mathbf{x}}(i,j) P(\theta_{\mathbf{x}}|\mathbf{x}).$$

For $\{L_{\mathbf{x}}(i), R_{\mathbf{x}}(i), U_{\mathbf{x}}(i)\}$, we obtain three kinds of probability that the i-th position is paired with one of the downstream/upstream positions, or unpaired, respectively:

$$\begin{aligned}
P^L_{\mathbf{x}}(i) &= \sum_{\theta_{\mathbf{x}} \in \Theta_{\mathbf{x}}} L_{\mathbf{x}}(i) P(\theta_{\mathbf{x}}|\mathbf{x}) = \sum_{\theta_{\mathbf{x}} \in \Theta_{\mathbf{x}}} \sum_{j:j>i} \theta_{\mathbf{x}}(i,j) P(\theta_{\mathbf{x}}|\mathbf{x}) = \sum_{j:j>i} P_{\mathbf{x}}(i,j), \\
P^R_{\mathbf{x}}(i) &= \sum_{\theta_{\mathbf{x}} \in \Theta_{\mathbf{x}}} R_{\mathbf{x}}(i) P(\theta_{\mathbf{x}}|\mathbf{x}) = \sum_{\theta_{\mathbf{x}} \in \Theta_{\mathbf{x}}} \sum_{j:j<i} \theta_{\mathbf{x}}(j,i) P(\theta_{\mathbf{x}}|\mathbf{x}) = \sum_{j:j<i} P_{\mathbf{x}}(j,i), \\
P^U_{\mathbf{x}}(i) &= \sum_{\theta_{\mathbf{x}} \in \Theta_{\mathbf{x}}} U_{\mathbf{x}}(i) P(\theta_{\mathbf{x}}|\mathbf{x}) = 1 - P^L_{\mathbf{x}}(i) - P^R_{\mathbf{x}}(i).
\end{aligned}$$

We design a scoring function $S_{\mathbf{xy}}(i,j)$ by taking the expectation of (Eq. 3.3) over $\Theta_{\mathbf{x}}$ and $\Theta_{\mathbf{y}}$:

$$\begin{aligned}
S_{\mathbf{xy}}(i,j) &= \sum_{\theta_{\mathbf{x}} \in \Theta_{\mathbf{x}}} \sum_{\theta_{\mathbf{y}} \in \Theta_{\mathbf{y}}} W_{\mathbf{xy}}(i,j|\theta_{\mathbf{x}}, \theta_{\mathbf{y}}) P(\theta_{\mathbf{x}}|\mathbf{x}) P(\theta_{\mathbf{y}}|\mathbf{y}) \\
&= \alpha \left(P^L_{\mathbf{x}}(i) P^L_{\mathbf{y}}(j) + P^R_{\mathbf{x}}(i) P^R_{\mathbf{y}}(j) \right) \\
&\quad + s(x_i, y_j) P^U_{\mathbf{x}}(i) P^U_{\mathbf{y}}(j). \qquad \text{(Eq. 3.4)}
\end{aligned}$$

The proposed method is obtained by combining the normalized LA kernel (Eq. 3.2) with the scoring function (Eq. 3.4).

It should be noted that our method can take into account all possible secondary structures in $O(|\mathbf{x}|^3 + |\mathbf{y}|^3)$ time, owing to the McCaskill algorithm. Just as in all possible sequence alignments, the exact Sankoff algorithm results in $O(|\mathbf{x}|^3|\mathbf{y}|^3)$ time, and the existing methods cannot incorporate all possible secondary structures. Our method requires $O(|\mathbf{x}||\mathbf{y}|)+O(|\mathbf{x}|^3+|\mathbf{y}|^3)$ time in total, which is more efficient than the exact Sankoff algorithm. Therefore, our strategy that combines the SW algorithm and the McCaskil algorithm allows to utilize the ensemble information with the reasonable computational cost.

3.2.3 Variations of the proposed method

The scoring function (Eq. 3.4) proposed in this study is similar to the scoring function used in BPLA kernel (Morita *et al.*, 2009; Dalli *et al.*, 2006). BPLA kernel is a prediction method that we previously developed for detecting new members of known ncRNA families. Although BPLA kernel was not applied to clustering problems in our previous study, we here clarify its relation to the proposed method. The scoring

function used in BPLA kernel is defined as follows:

$$\begin{aligned} S^{\mathrm{BPLA}}_{\mathbf{xy}}(i,j) &= \alpha\left(\sqrt{P^L_{\mathbf{x}}(i)P^L_{\mathbf{y}}(j)} + \sqrt{P^R_{\mathbf{x}}(i)P^R_{\mathbf{y}}(j)}\right) \\ &\quad + s(x_i, y_j)\sqrt{P^U_{\mathbf{x}}(i)P^U_{\mathbf{y}}(j)} \\ &= \alpha\left(C^L P^L_{\mathbf{x}}(i)P^L_{\mathbf{y}}(j) + C^R P^R_{\mathbf{x}}(i)P^R_{\mathbf{y}}(j)\right) \\ &\quad + s(x_i, y_j) C^U P^U_{\mathbf{x}}(i)P^U_{\mathbf{y}}(j), \end{aligned} \qquad \text{(Eq. 3.5)}$$

where $C^L = 1/\sqrt{P^L_{\mathbf{x}}(i)P^L_{\mathbf{y}}(j)}$, $C^R = 1/\sqrt{P^R_{\mathbf{x}}(i)P^R_{\mathbf{y}}(j)}$, and $C^U = 1/\sqrt{P^U_{\mathbf{x}}(i)P^U_{\mathbf{y}}(j)}$. Therefore, the scoring function (Eq. 3.5) can be regarded as a variation of the proposed scoring function (Eq. 3.4) with the additional coefficients C^L, C^R, and C^U. These coefficients take large values when the probabilities $P^{\cdot}_{\mathbf{x}}(i)$ and $P^{\cdot}_{\mathbf{y}}(j)$ are small. That is, BPLA kernel emphasizes the contribution of low-probability (unsure) secondary structures compared to the proposed method. In the next section, we experimentally verify this theoretical implication; the proposed method outperforms BPLA kernel.

Because of the resemblance between the scoring functions, (Eq. 3.4) and (Eq. 3.5), we set the parameters of the proposed method as used in BPLA kernel: $\alpha = 1.0$, $\beta = 0.1$, $g = -27$, and $d = -0.1$

3.2.4 Availability

Our implementation of the proposed method is available for download at http://bpla-kernel.dna.bio.keio.ac.jp/clustering/ under the GNU general public license. It takes a set of RNA sequences in the MAF format, and produces a cluster tree in the Newick format, which can be visualized by the ape package for the R statistical computing environment (http://cran.r-project.org/web/packages/ape/).

3.3 Results and discussion

In this section, we examine the performance of the proposed method in the hierarchical clustering of ncRNAs.

3.3.1 Dataset and experimental system

We compared our method with the state-of-the-art methods developed for the hierarchical clustering of ncRNAs: LocARNA v1.5.2 (Will *et al.*, 2007), FOLDALIGN v2.1.1 (Havgaard *et al.*, 2007), and Stem kernel v216c (Sato *et al.*, 2008). We also performed the experiments with CLUSTALW v1.83 (Thompson *et al.*, 1994) and LA kernel by setting $\{P^L_{\mathbf{x}}(i) = 0, P^R_{\mathbf{x}}(i) = 0, P^U_{\mathbf{x}}(i) = 1\}$ in our method (Eq. 3.4).

We can summarize our method and the existing methods as follows. Our method utilizes *all possible* sequence alignments and *all possible* secondary structures. LocARNA and FOLDALIGN only use *one optimal* sequence alignment and *one optimal* secondary structures. Stem kernel utilizes a *subset of* all possible sequence

Table 3.1 Summary of the dataset.

	20–39%	40–59%	60–79%	80–99%
#clusters	13	21	34	36
#members	3.2	5.0	3.8	4.6
Length	138	130	111	102

#clusters: number of reference clusters; each reference cluster represents a different ncRNA family. #members: average number of member sequences per reference cluster. Length: average length of sequences over all reference clusters. The dataset is divided by the sequence identity in a reference cluster.

alignments and a *subset of* all possible secondary structures. CLUSTALW and LA kernel ignore secondary structures; CLUSTALW only uses *one optimal* sequence alignment, while LA kernel utilizes *all possible* sequence alignments.

We created a dataset as summarized in Table 3.1. This dataset was collected from the BRAliBASE benchmark v2.1 (Wilm *et al.*, 2006), which includes multiple alignments of a broad range of ncRNA families established in the Rfam database (Gardner *et al.*, 2009). We treated each multiple alignment as a reference cluster, and each ncRNA sequence in a multiple alignment as a member sequence. The reference clusters were divided into four categories according to their sequence identity: 20–39%, 40–59%, 60–79%, and 80–99%. We sampled the dataset ten times from the BRAliBASE benchmark, and evaluated the average performance.

We produced three versions of dataset. First, we used ncRNA sequences without modification, and named them the "normal" dataset. Second, we concatenated random sequences to both ends of ncRNA sequences, and named them the "plus flanking regions" dataset. This dataset was intended to simulate the situation where we do not know the exact boundaries of unannotated transcripts. A random sequence was generated from a ncRNA sequence so that it had the quarter length and the same dinucleotide contents. Third, we added false reference clusters, each of which contains one random sequence, and named them the "plus unrelated sequences" dataset. This dataset was intended to simulate the situation where non-functional ncRNAs arise from transcriptional noises. Therefore, we evaluated whether a false reference cluster could be a resultant cluster with a single member. We used a quarter number of false reference clusters compared to true reference clusters. A random sequence was generated from a ncRNA sequence so that it had the same length and the same dinucleotide contents.

We evaluated the overall quality of the cluster tree by the ROC analysis proposed in (Will *et al.*, 2007). (Note that we can obtain different resultant clusters from a cluster tree depending on a distance threshold to cut the branches.) Given a distance threshold, the number of true positives (TP) was defined as the number of sequence pairs that belong to the same reference cluster and are correctly assigned to the same resultant cluster. Analogously, the numbers of false positives (FP), true negatives (TN), and false negatives (FN) are defined, respectively, by counting the pairs from different reference clusters but the same resultant cluster, the pairs

from different reference clusters and different resultant clusters, and the pairs from the same reference cluster but different resultant clusters. The ROC analysis was performed by plotting true positive rates $TP/(TP + FN)$ versus false positive rates $FP/(TN + FP)$ for different distance thresholds. The quality of the clustering was measured by the area under the ROC curve (AUC). We measured the total time for computing similarity matrices on a 2.53 GHz Intel Xeon processor.

Our experiments aimed to evaluate the performance of clustering methods by reconstructing known families from a set of single sequences. This might sound somewhat strange because the purpose of clustering approaches is to find novel families rather than known families. However, we emphasize that known families to be reconstructed were virtually treated as novel families throughout our experiments. The proposed method and the other existing methods do not employ any feature values specifically adjusted to known families. In addition, these methods do not use training data such as member sequences of families to be detected. Therefore, the performance of each method observed in our experiments should be applied to the problem of finding (completely) unknown families.

3.3.2 Quality of the clustering

We first examined the quality of the clustering for the "normal" dataset (Figure 3.1). Our method achieved the better or comparable AUC to the existing methods in all the range of sequence identity. The accuracy of our method was especially remarkable in the sequence identity range below 60%, where the existing methods resulted in low AUC. This means that our method successfully grouped diverse member sequences in each reference cluster by detecting their remote homology.

Our results can be attributed to the design of each method. The AUC of CLUSTALW and LA kernel, which ignore secondary structures and only use sequence alignments, drastically fell down as the sequence identity decreased. LocARNA, FOLDALIGN, and Stem kernel, which consider secondary structures, kept the AUC relatively moderate in the low sequence identity range. However, their accuracy was still limited when the sequence identity was extremely low (20–39%) because these methods only use *one optimal* secondary structure or a *subset of* secondary structures. Our method, which utilizes *all possible* sequence alignments and *all possible* secondary structures, achieved the sufficiently high AUC in this region. These results suggest that our design of the similarity measure is effective for identifying a broad range of ncRNA families.

We found that the AUC of FOLDALIGN in the sequence identity range of 40–59% was substantially better than LocARNA and Stem kernel, being comparable to the proposed method. FOLDALIGN is different from LocARNA and Stem kernel in its heuristics to choose the *one optimal* structural alignment from the DP matrix. LocARNA and Stem kernel preclude unsure structural alignments based on secondary structure prediction of each sequence to be aligned, and prepare the reduced DP matrix before the computation of structural alignment. On the other hand, FOLDALIGN initiates the computation of structural alignment with the

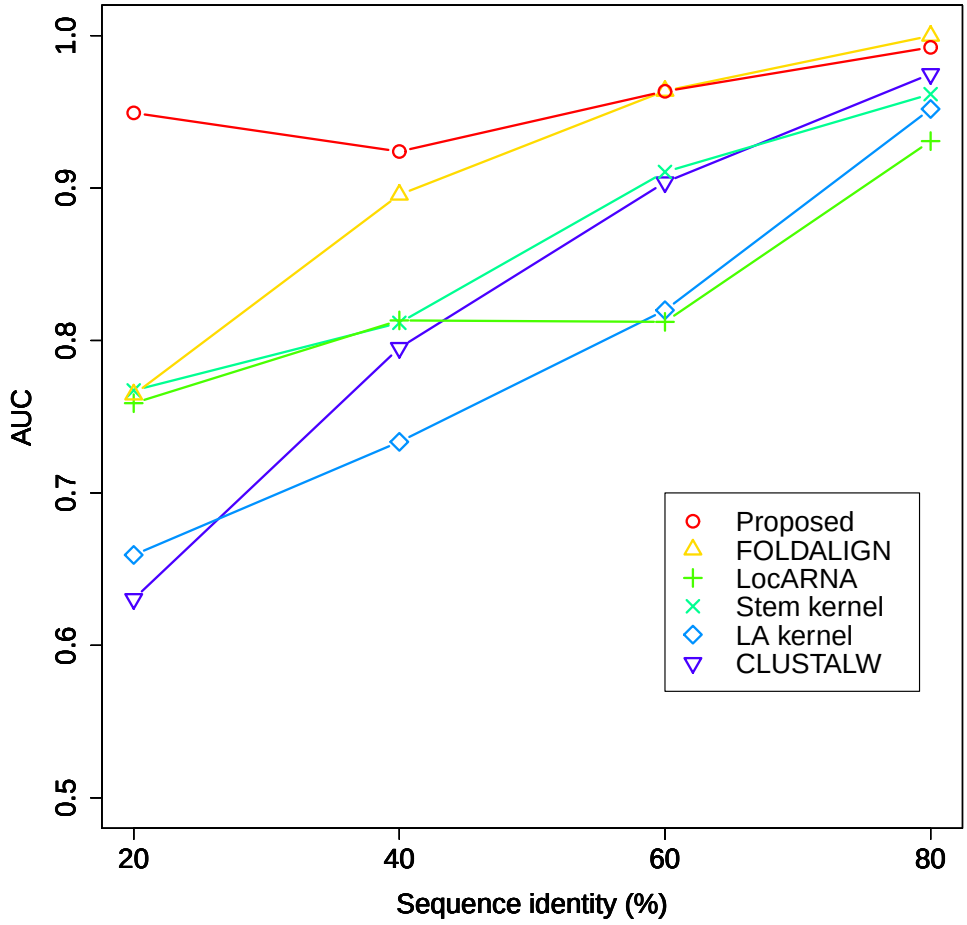


Figure 3.1 Quality of the clustering for the "normal" dataset. For each sequence identity range, the overall quality of the cluster tree is evaluated by the AUC.

full-size DP matrix as used in the original Sankoff algorithm. Then, FOLDALIGN dynamically excludes low-scoring structural alignments along with the computation by discarding the portion of the DP matrix by means of length-dependent thresholds. We consider that the heuristics in FOLDALIGN might be suitable for the low sequence identity range because remote homology due to the covariation of base pairs can only be detected by aligning these base pairs. Nevertheless, the serious drawback of this strategy is that it is impossible to incorporate the structural alignments in the excluded portion of the DP matrix that might be useful for evaluating the similarity. In fact, the AUC of FOLDALIGN in the sequence identity range of 40–59% was slightly worse than the proposed method that is designed to incorporate *all possible* sequence alignments and *all possible* secondary structures. Furthermore, the difference in AUC became more remarkable in the sequence identity range of 20–39%.

These results suggest that suboptimal structural alignments, which are discarded by FOLDALIGN but utilized by the proposed method, have the useful information to improve the quality of clustering.

Figure 3.2 compares an example of the cluster tree between our method and FOLDALIGN in the sequence identity rage of 20–39%. As indicated by AUC, our method produced the more accurate cluster tree than FOLDALIGN, and reconstructed ncRNA families as compact clusters. Although the cluster tree of FOLDALIGN was largely consistent with the references in terms of its topology, boundaries of resultant clusters were quite unclear. In the actual application of hierarchical clustering, we need to choose a proper distance threshold for extracting clusters from a given tree. In this sense, the cluster tree of FOLDALIGN was not sufficient for the practical use. In fact, the previous studies that employed clustering approaches required manual inspection to compensate for ambiguous cluster trees (Will *et al.*, 2007; Weinberg *et al.*, 2009, 2010). The cluster tree of our method was much more clear and easier to interpret than the existing methods. These results suggest that our method can reduce human labor costs of clustering approaches, and help to identify novel ncRNAs families.

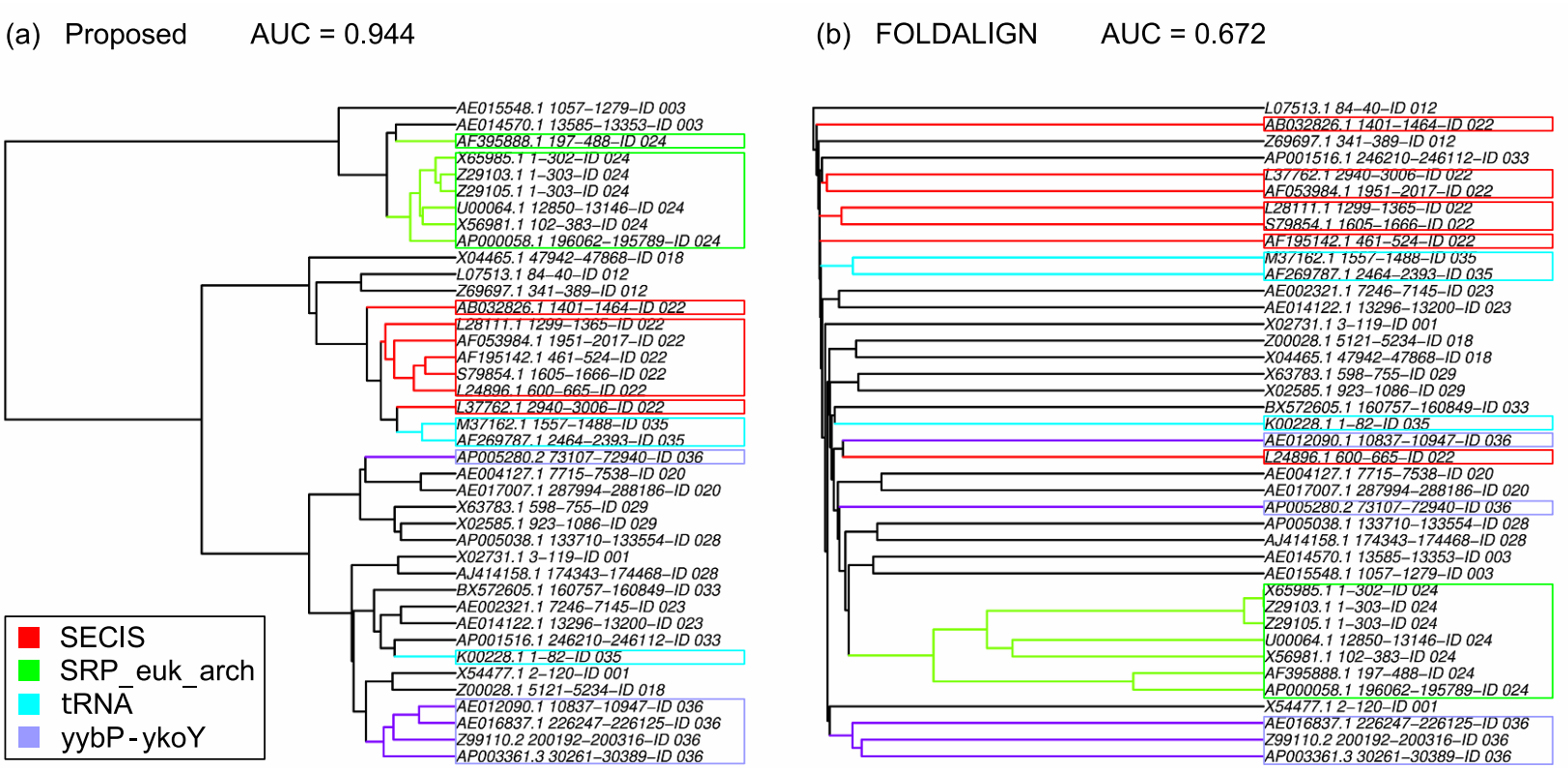


Figure 3.2 Comparison of the cluster trees between the proposed method and FOLDALIGN. For the sequence identity range of 20–39% in the "normal" dataset, an example of the cluster tree is shown with its AUC. In the leaf nodes, the strings such as "`XXX-ID YYY`" mean that the sequence `XXX` belongs to the reference cluster `YYY`. The four reference clusters that have more than two member sequences are colored, and their corresponding ncRNA families are noted.

Table 3.2 Computational cost of the similarity measures.

	Computation time (s)		
Method	Normal	Plus flanking regions	Plus unrelated sequences
Proposed	95	222	199
FOLDALIGN	71748	226066	167228
LocARNA	9704	64679	30287
Stem kernel	61	179	138
LA kernel	71	163	160
CLUSTALW	4	43	6

The total time for computing similarity matrices is shown for three versions of the dataset.

Next, we evaluated the quality of the clustering for the "plus flanking regions" dataset (Figure 3.3), and the "plus unrelated sequences" dataset (Figure 3.4). In both cases, we observed the same tendency as in the results for the "normal" dataset (Figure 3.1). Our method kept high accuracy in all the range of sequence identity, and achieved the best AUC in the sequence identity range below 60%. These results further support the effectiveness of our method in the practical situations that involve flanking regions and unrelated sequences.

3.3.3 Differences in the variations of the proposed method

As described in Methods, the proposed method has the theoretical advantage compared to BPLA kernel, which can be regarded as a variation of our method. To verify this point experimentally, we compare the proposed method and BPLA kernel using the scoring functions (Eq. 3.4) and (Eq. 3.5), respectively.

Figure 3.5 presents the experimental results. The proposed method achieved the slightly better AUC in the sequence identity range below 60%. These results are consistent with the fact that BPLA kernel emphasizes the contribution of unsure secondary structures compared to the proposed method. The proposed scoring function (Eq. 3.4) has the theoretical justification as the expectation of the primitive scoring function (Eq. 3.3) over all possible secondary structures. Our results provide an experimental verification of the superiority of the proposed scoring function.

3.3.4 Computational cost

Finally, we evaluated the computational cost of the similarity measures using three version of the dataset (Table 3.2). Our method was faster than LocARNA and FOLDALIGN by several orders of magnitude, and achieved the comparable computational cost to Stem kernel. Considering the high accuracy of our method (Figures 3.1–3.4), we achieved the best balance between the computational cost and the quality of the clustering among the existing methods.

In the design of the proposed method, our idea was to improve the reliability of approximate algorithms by the information of suboptimal solutions in their DP frameworks. Among LocARNA and FOLDALIGN, which only use *one optimal*

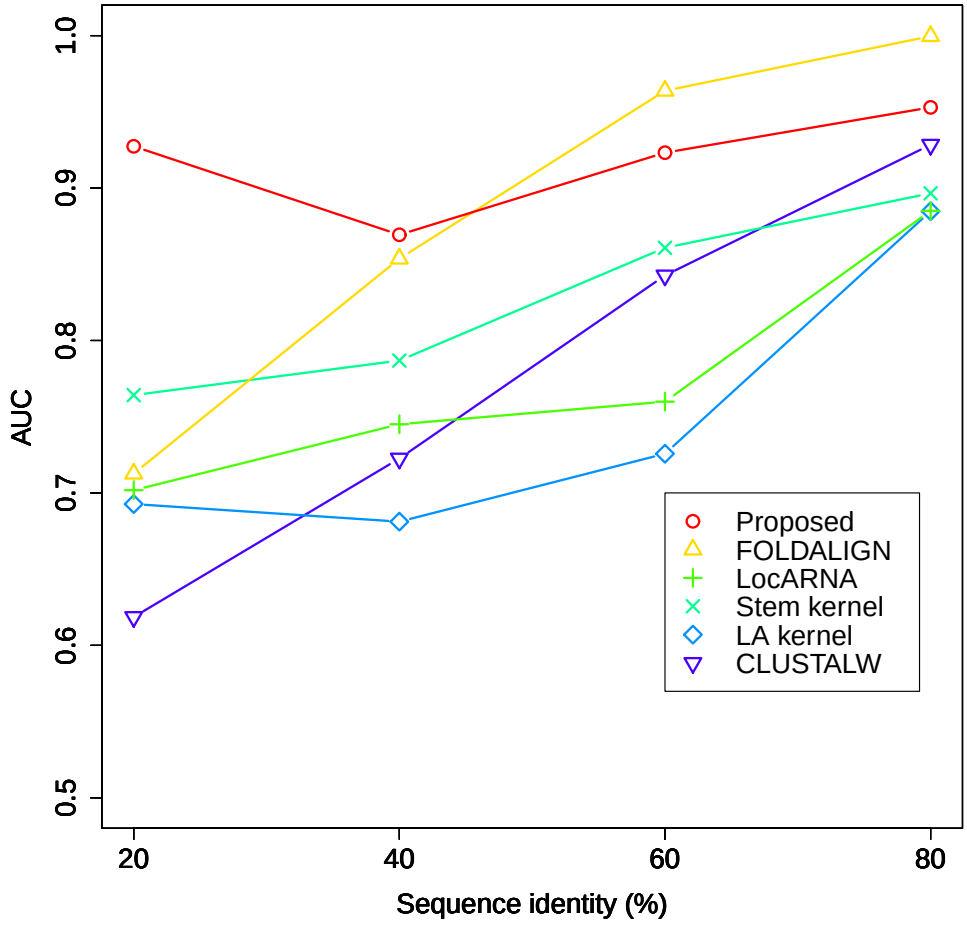


Figure 3.3 Quality of the clustering for the "plus flanking regions" dataset. For each sequence identity range, the overall quality of the cluster tree is evaluated by the AUC.

solution in their approximate Sankoff-style algorithms, there was a trade-off that LocARNA was faster but less accurate than FOLDALIGN (Figure 3.1, and Table 3.2). Stem kernel, which utilizes a *subset of* solutions in the more approximate Sankoff-style algorithm, partly improved this problem, being faster and more accurate than LocARNA. Our method, which utilizes *all possible* solutions in the combination of the Smith-Waterman algorithm and the McCaskill algorithm, successfully overcome the trade-off. These results suggest that our strategy is essential to enable fast and accurate clustering of ncRNAs.

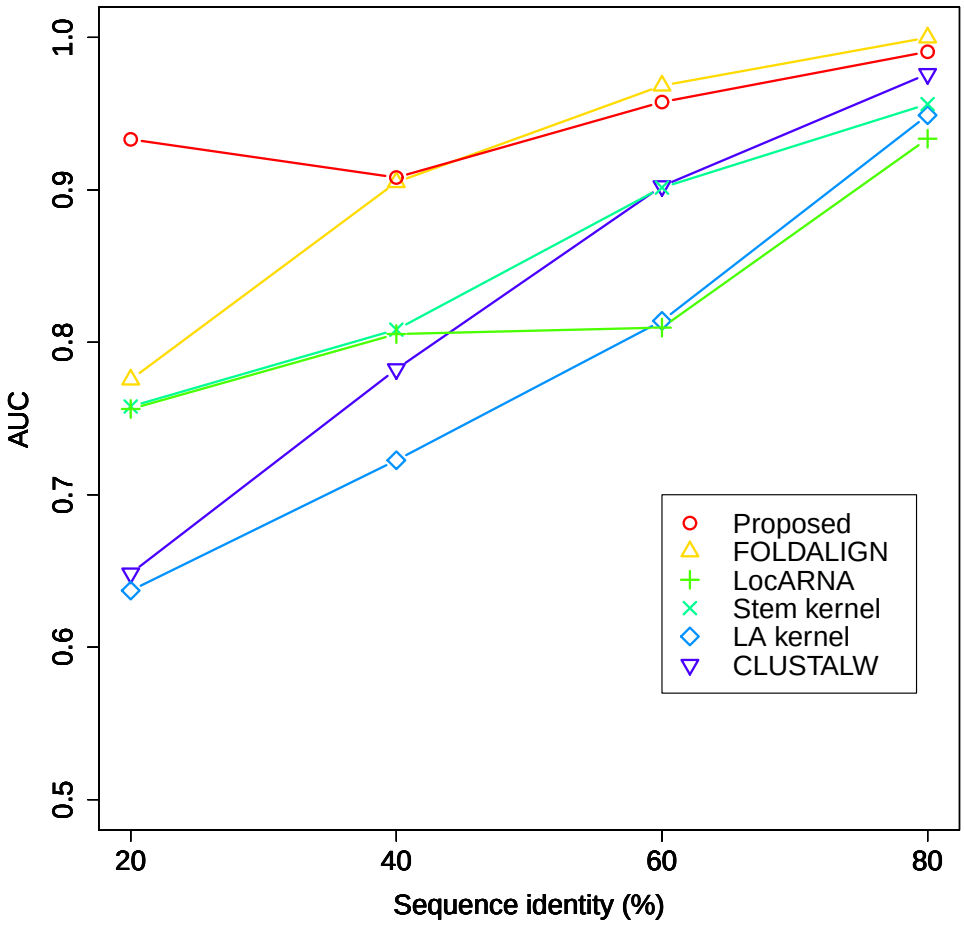


Figure 3.4 Quality of the clustering for the "plus unrelated sequences" dataset. For each sequence identity range, the overall quality of the cluster tree is evaluated by the AUC.

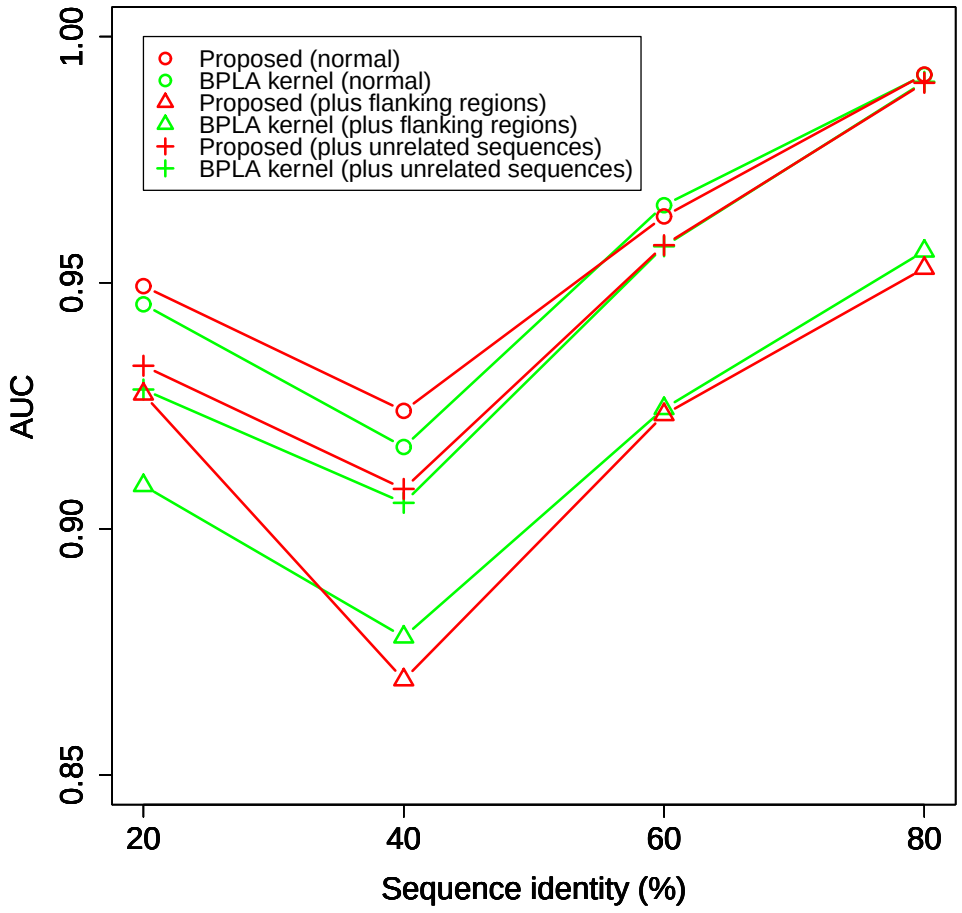


Figure 3.5 Differences in the variations of the proposed method. The proposed method is compared to BPLA kernel using three versions of the dataset. Note that BPLA kernel can be regarded as a variation of the proposed method.

3.4 Conclusions

We have described a new method for the hierarchical clustering of ncRNAs, which can be applied to the identification of novel ncRNA families. Our method can achieve the best balance between the computational cost and the quality of the clustering compared to the existing methods.

The performance of the clustering is determined by similarity measures based on the scores of structural alignment. The existing similarity measures, which only use *one optimal* structural alignment, suffer from the trade-off between time-consuming accurate computation and fast approximate computation. Our similarity measure, which is designed to utilize *all possible* sequence alignments and *all possible* secondary structures, have overcome this problem. The improvement is especially remarkable when the similarity among family members is not detectable at the primary sequence level.

In conclusion, our method enables fast and accurate clustering of ncRNAs, providing a promising way to explore the functional diversity of ncRNAs.

Chapter 4

Conclusion and future work

In this dissertation, we have presented two computational methods for the identification of noncoding RNAs (ncRNAs). The first method has been proposed for predicting whether an input RNA is a new member of a known ncRNA family. The second method has been proposed for finding candidates of novel ncRNA families from a set of unannotated RNAs. Both of the proposed methods outperformed the previous state-of-the-art methods developed for the same purposes.

Our methods can be regarded as applications of similarity search, and the originality of our studies is the design of similarity measures for ncRNAs. In Chapter 2, we have described Profile BPLA kernel that measures the similarity between two alignment data of RNAs by utilizing the profile information. This similarity measure enables to predict ncRNAs from alignment data in combination with SVMs. In Chapter 3, we have described a similarity measure between two RNA sequences that utilizes the ensemble information.

Similarity search is a fundamental task in biological sequence analysis, and thus has a wide range of applications other than those addressed in this dissertation. For example, database search is a common application of similarity search where a similarity measure is evaluated between a query and each of database entries. In addition, the construction of genome alignment involve a procedure in which a similarity measure is evaluated among all pairs of short segments in genomic sequences. Our similarity measures can be used in these applications, and may improve the performance of existing frameworks for ncRNAs. Especially, genome-wide alignment using secondary structure information has been one of the unsolved problems in RNA informatics (Torarinsson *et al.*, 2006, 2008). Accurate genome alignments of ncRNAs can improve a series of downstream analyses such as prediction of consensus secondary structures, and evaluation of structure conservation. Moreover, some RNA viruses such as human immunodeficiency virus-1 (HIV-1) have an RNA genome which can be considered as a huge structural RNA (Watts *et al.*, 2009). Aligning these RNA genomes using secondary structure information may provide insights into the virus evolution at the level of secondary structures.

The similarity search methods proposed in our studies are *brute-force* in the sense that these methods require to evaluate a similarity measure for all RNAs in a dataset. For example, the first method needs to calculate Profile BPLA kernel between training samples and all of input alignment data. The second method also needs to calculate its similarity measure between all pairs in a given set. Although our methods are relatively efficient among existing methods, which are also brute-force, the computational cost will be prohibitive for extremely large datasets. To address this issue, we are now planning to develop more efficient search algorithms based on *index-based* techniques (Figure 4.1). BLAST (Altschul *et al.*, 1990) is a widely-used database search tool that employ the Smith-Waterman (SW) alignment score as a

similarity measure, but avoid brute-force search by using an index-based technique. In the method, database sequences are stored in an index which enables to find short consecutive matches of nucleotide strings shared with a query. BLAST uses these matches as candidates, and evaluates the SW score only for database entries sharing the matches. As a consequence of this strategy, BLAST can achieve efficient search even for extremely large datasets. However, the same strategy cannot be directly applied to our similarity measures, because we cannot build an index for secondary structure information represented by continuous probability values rather than string data like nucleotide sequences. Therefore, we are planning to employ another strategy called locality-sensitive hashing (LSH) (Indyk and Motwani, 1998), which recently draws much attention in the domain of information retrieval. Index-based methods that can incorporate secondary structure information has been one of the major goals in RNA informatics. Index-based method for our similarity measures may realize this goal.

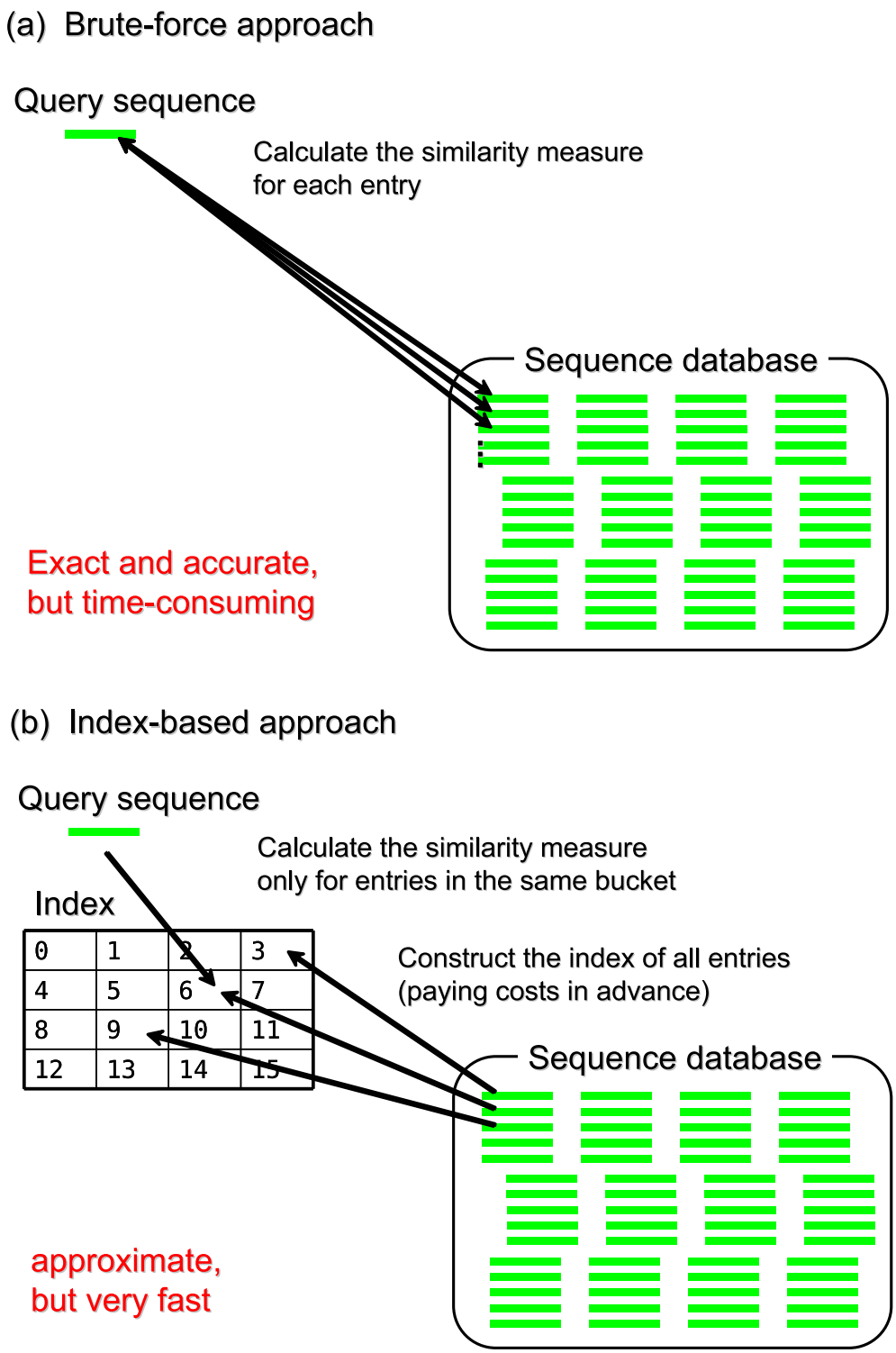


Figure 4.1 Comparison of the brute-force approach and the index-based approach in a database search problem. (a) Brute-force approach. A query is directly compared to each of all entries in a database using a certain similarity measure. (b) Index-based approach. A database is converted in advance so that candidates of high-scoring entries for a query can be found efficiently.

Acknowledgements

First of all, I would like to express my sincere thanks to Professor Yasubumi Sakakibara who has provided the comprehensive support for my study as a supervisor of my bachelor, master, and doctor courses. Especially, I appreciate that he kindly gave me a lot of opportunities for presenting my study in journal papers and international conferences. These valuable experiences laid the foundation of my ability, and helped me make up my mind to become a professional researcher.

I also would like to express my cordial gratitude to Assistant Professor Kengo Sato who has provided a number of technical advice for my study as a leading researcher in RNA informatics. He and his colleagues in Computational Biology Research Center developed state-of-the-art methods in this domain, and such an atmosphere was highly stimulating to me. It is not an exaggeration to say that I learned RNA informatics from his paper, and learned programming languages from his source code.

I am very grateful to my colleagues in Sakakibara Laboratory, for making a good academic environment. I would like to thank Assistant Professor Katsuyuki Yugi for teaching me how to write scientific papers. I also would like to thank Dr. Yasunori Osana, Dr. Tsuyoshi Hachiya, and Mr. Kris Popendorf for giving me critical comments on my study as the senior members in the laboratory. My thanks also go to the members of the RNA research team, including Mr. Kensuke Morita, Mr. Youhei Okada, Ms. Junko Kawarama, Mr. Masahiro Ogawa, Mr. Masaya Abe, and Mr. Shunya Kashiwagi.

I wish to thank Yoshida Scholarship Foundation that has provided the generous financial support for my school and living expenses, without which I could not complete my doctoral program. I have enormous respect for the spirit and the philosophy of the foundation.

Lastly, I would like to express my sincere thanks to Professor Yasubumi Sakakibara, Professor Akio Kanai, Professor Kotaro Oka, and Associate Professor Nobuhide Doi for examining and judging my doctoral dissertation.

References

Aiba, H. Mechanism of RNA silencing by Hfq-binding small RNAs. *Curr. Opin. Microbiol.*, 10(2): 134–139, 2007.

Altschul, S. F. and Erickson, B. W. Significance of nucleotide sequence alignments: a method for random sequence permutation that preserves dinucleotide and codon usage. *Mol. Biol. Evol.*, 2 (6):526–538, 1985.

Altschul, S. F., Gish, W., Miller, W., Myers, E. W., and Lipman, D. J. Basic local alignment search tool. *J. Mol. Biol.*, 215(3):403–410, 1990.

Backofen, R., Bernhart, S. H., Flamm, C., Fried, C., Fritzsch, G., Hackermuller, J., Hertel, J., Hofacker, I. L., Missal, K., Mosig, A., Prohaska, S. J., Rose, D., Stadler, P. F., Tanzer, A., Washietl, S., and Will, S. RNAs everywhere: genome-wide annotation of structured RNAs. *J. Exp. Zool. B Mol. Dev. Evol.*, 308(1):1–25, 2007.

Bernhart, S. H., Hofacker, I. L., Will, S., Gruber, A. R., and Stadler, P. F. RNAalifold: improved consensus structure prediction for RNA alignments. *BMC Bioinformatics*, 9:474, 2008.

Bonhoeffer, S., McCaskill, J. S., Stadler, P. F., and Schuster, P. RNA multi-structure landscapes. A study based on temperature dependent partition functions. *Eur. Biophys. J.*, 22(1):13–24, 1993.

Boser, B. E., Guyon, I. M., and Vapnik, V. N. A training algorithm for optimal margin classifiers. In *Proceedings of the 5th Annual ACM Workshop on Computational Learning Theory*, pages 144–152. ACM Press, 1992.

Dalli, D., Wilm, A., Mainz, I., and Steger, G. STRAL: progressive alignment of non-coding RNA using base pairing probability vectors in quadratic time. *Bioinformatics*, 22(13):1593–1599, 2006.

Dambach, M. D. and Winkler, W. C. Expanding roles for metabolite-sensing regulatory RNAs. *Curr. Opin. Microbiol.*, 12(2):161–169, 2009.

Do, C. B., Foo, C. S., and Batzoglou, S. A max-margin model for efficient simultaneous alignment and folding of RNA sequences. *Bioinformatics*, 24(13):68–76, 2008.

Dulebohn, D., Choy, J., Sundermeier, T., Okan, N., and Karzai, A. W. Trans-translation: the tmRNA-mediated surveillance mechanism for ribosome rescue, directed protein degradation, and nonstop mRNA decay. *Biochemistry*, 46(16):4681–4693, 2007.

Eddy, S. R. Profile hidden Markov models. *Bioinformatics*, 14(9):755–763, 1998.

Eddy, S. R. Non-coding RNA genes and the modern RNA world. *Nat. Rev. Genet.*, 2(12):919–929, 2001.

Eddy, S. R. Computational genomics of noncoding RNA genes. *Cell*, 109(2):137–140, 2002.

Fan, R. E., Chen, P. H., and Lin, C. J. Working set selection using second order information for training support vector machines. *Journal of Machine Learning Research*, 6:1889–1918, 2005.

Filipowicz, W., Bhattacharyya, S. N., and Sonenberg, N. Mechanisms of post-transcriptional regulation by microRNAs: are the answers in sight? *Nat. Rev. Genet.*, 9(2):102–114, 2008.

Gardner, P. P., Daub, J., Tate, J. G., Nawrocki, E. P., Kolbe, D. L., Lindgreen, S., Wilkinson, A. C., Finn, R. D., Griffiths-Jones, S., Eddy, S. R., and Bateman, A. Rfam: updates to the RNA families database. *Nucleic Acids Res.*, 37(Database issue):D136–D140, 2009.

Gardner, P. P., Daub, J., Tate, J., Moore, B. L., Osuch, I. H., Griffiths-Jones, S., Finn, R. D., Nawrocki, E. P., Kolbe, D. L., Eddy, S. R., and Bateman, A. Rfam: Wikipedia, clans and the "decimal" release. *Nucleic Acids Res.*, 39(Database issue):D141–D145, 2011.

Gesell, T. and Washietl, S. Dinucleotide controlled null models for comparative RNA gene prediction. *BMC Bioinformatics*, 9:248, 2008.

Gruber, A. R., Findeiss, S., Washietl, S., Hofacker, I. L., and Stadler, P. F. RNAZ 2.0: IMPROVED NONCODING RNA DETECTION. *Pac. Symp. Biocomput.*, 15:69–79, 2010.

Guttman, M., Garber, M., Levin, J. Z., Donaghey, J., Robinson, J., Adiconis, X., Fan, L., Koziol, M. J., Gnirke, A., Nusbaum, C., Rinn, J. L., Lander, E. S., and Regev, A. Ab initio reconstruction of cell type-specific transcriptomes in mouse reveals the conserved multi-exonic structure of lincRNAs. *Nat. Biotechnol.*, 28(5):503–510, 2010.

Hamada, M., Kiryu, H., Sato, K., Mituyama, T., and Asai, K. Prediction of RNA secondary structure using generalized centroid estimators. *Bioinformatics*, 25(4):465–473, 2009.

Harley, C. B. Telomerase and cancer therapeutics. *Nat. Rev. Cancer*, 8(3):167–179, 2008.

Havgaard, J. H., Torarinsson, E., and Gorodkin, J. Fast pairwise structural RNA alignments by pruning of the dynamical programming matrix. *PLoS Comput. Biol.*, 3(10):e193, 2007.

Hofacker, I. L. Vienna RNA secondary structure server. *Nucleic Acids Res.*, 31(13):3429–3431, 2003.

Hüttenhofer, A., Schattner, P., and Polacek, N. Non-coding RNAs: hope or hype? *Trends Genet.*, 21(5):289–297, 2005.

Indyk, P. and Motwani, R. Approximate nearest neighbors: towards removing the curse of dimensionality. In *Proceedings of the 30th annual ACM symposium on Theory of computing*, pages 604–613. ACM press, 1998.

Kato, Y., Sato, K., Hamada, M., Watanabe, Y., Asai, K., and Akutsu, T. RactIP: fast and accurate prediction of RNA-RNA interaction using integer programming. *Bioinformatics*, 26(18):i460–i466, 2010.

Kazan, H., Ray, D., Chan, E. T., Hughes, T. R., and Morris, Q. RNAcontext: a new method for learning the sequence and structure binding preferences of RNA-binding proteins. *PLoS Comput. Biol.*, 6(7):e1000832, 2010.

Kim, V. N., Han, J., and Siomi, M. C. Biogenesis of small RNAs in animals. *Nat. Rev. Mol. Cell Biol.*, 10(2):126–139, 2009.

Kiryu, H., Kin, T., and Asai, K. Robust prediction of consensus secondary structures using averaged base pairing probability matrices. *Bioinformatics*, 23(4):434–441, 2007.

Kiss, T. Small nucleolar RNA-guided post-transcriptional modification of cellular RNAs. *EMBO J.*, 20(14):3617–3622, 2001.

Klein, R. J. and Eddy, S. R. RSEARCH: finding homologs of single structured RNA sequences. *BMC Bioinformatics*, 4:44, 2003.

Kuhn, R. M., Karolchik, D., Zweig, A. S., Wang, T., Smith, K. E., Rosenbloom, K. R., Rhead, B., Raney, B. J., Pohl, A., Pheasant, M., Meyer, L., Hsu, F., Hinrichs, A. S., Harte, R. A., Giardine, B., Fujita, P., Diekhans, M., Dreszer, T., Clawson, H., Barber, G. P., Haussler, D., and Kent, W. J. The UCSC Genome Browser Database: update 2009. *Nucleic Acids Res.*, 37(Database issue):D755–D761, 2009.

Lapidot, M. and Pilpel, Y. Genome-wide natural antisense transcription: coupling its regulation to its different regulatory mechanisms. *EMBO Rep.*, 7(12):1216–1222, 2006.

Lee, Y. S., Shibata, Y., Malhotra, A., and Dutta, A. A novel class of small RNAs: tRNA-derived RNA fragments (tRFs). *Genes Dev.*, 23(22):2639–2649, 2009.

Lukavsky, P. J. Structure and function of HCV IRES domains. *Virus Res.*, 139(2):166–171, 2009.

Majdalani, N., Vanderpool, C. K., and Gottesman, S. Bacterial small RNA regulators. *Crit. Rev. Biochem. Mol. Biol.*, 40(2):93–113, 2005.

Mallatt, J. and Winchell, C. J. Ribosomal RNA genes and deuterostome phylogeny revisited: more cyclostomes, elasmobranchs, reptiles, and a brittle star. *Mol. Phylogenet. Evol.*, 43(3):1005–1022, 2007.

Malone, C. D. and Hannon, G. J. Small RNAs as guardians of the genome. *Cell*, 136(4):656–668, 2009.

McCaskill, J. S. The equilibrium partition function and base pair binding probabilities for RNA secondary structure. *Biopolymers*, 29(6-7):1105–1119, 1990.

Morita, K., Saito, Y., Sato, K., Oka, K., Hotta, K., and Sakakibara, Y. Genome-wide searching with base-pairing kernel functions for noncoding RNAs: computational and expression analysis of snoRNA families in Caenorhabditis elegans. *Nucleic Acids Res.*, 37(3):999–1009, 2009.

Nawrocki, E. P., Kolbe, D. L., and Eddy, S. R. Infernal 1.0: inference of RNA alignments. *Bioinformatics*, 25(10):1335–1337, 2009.

Ogle, J. M., Carter, A. P., and Ramakrishnan, V. Insights into the decoding mechanism from recent ribosome structures. *Trends Biochem. Sci.*, 28(5):259–266, 2003.

Pacheco, P. *Parallel Programming with MPI*. Morgan Kaufmann, San Francisco, 1996.

Parisien, M. and Major, F. The MC-Fold and MC-Sym pipeline infers RNA structure from sequence data. *Nature*, 452(7183):51–55, 2008.

Perreault, J., Perreault, J. P., and Boire, G. Ro-associated Y RNAs in metazoans: evolution and diversification. *Mol. Biol. Evol.*, 24(8):1678–1689, 2007.

Persson, H., Kvist, A., Vallon-Christersson, J., Medstrand, P., Borg, A., and Rovira, C. The non-

coding RNA of the multidrug resistance-linked vault particle encodes multiple regulatory small RNAs. *Nat. Cell Biol.*, 11(10):1268–1271, 2009.

Prakash, A. and Tompa, M. Measuring the accuracy of genome-size multiple alignments. *Genome Biol.*, 8(6):R124, 2007.

Punta, M., Coggill, P. C., Eberhardt, R. Y., Mistry, J., Tate, J., Boursnell, C., Pang, N., Forslund, K., Ceric, G., Clements, J., Heger, A., Holm, L., Sonnhammer, E. L., Eddy, S. R., Bateman, A., and Finn, R. D. The Pfam protein families database. *Nucleic Acids Res.*, 40(Database issue): D290–D301, 2012.

Rederstorff, M., Bernhart, S. H., Tanzer, A., Zywicki, M., Perfler, K., Lukasser, M., Hofacker, I. L., and Hüttenhofer, A. RNPomics: defining the ncRNA transcriptome by cDNA library generation from ribonucleo-protein particles. *Nucleic Acids Res.*, 38(10):e113, 2010.

Rosenblad, M. A., Larsen, N., Samuelsson, T., and Zwieb, C. Kinship in the SRP RNA family. *RNA Biol*, 6(5):508–516, 2009.

Saigo, H., Vert, J. P., Ueda, N., and Akutsu, T. Protein homology detection using string alignment kernels. *Bioinformatics*, 20(11):1682–1689, 2004.

Saito, Y., Sato, K., and Sakakibara, Y. Robust and accurate prediction of noncoding RNAs from aligned sequences. *BMC Bioinformatics*, 11(Suppl 7):S3, 2010.

Saito, Y., Sato, K., and Sakakibara, Y. Fast and accurate clustering of noncoding RNAs using ensembles of sequence alignments and secondary structures. *BMC Bioinformatics*, 12(Suppl 1): S48, 2011.

Sakakibara, Y., Popendorf, K., Ogawa, N., Asai, K., and Sato, K. Stem kernels for RNA sequence analyses. *J. Bioinform. Comput. Biol.*, 5(5):1103–1122, 2007.

Sankoff, D. Simultaneous solution of the RNA folding, alignment, and proto-sequence problems. *SIAM J. Appl Math*, 45(5):810–825, 1985.

Sato, K., Mituyama, T., Asai, K., and Sakakibara, Y. Directed acyclic graph kernels for structural RNA analysis. *BMC Bioinformatics*, 9:318, 2008.

Sato, K., Saito, Y., and Sakakibara, Y. Gradient-based optimization of hyperparameters for base-pairing profile local alignment kernels. *Genome Inform.*, 23(1):128–138, 2009.

Serganov, A. and Patel, D. J. Ribozymes, riboswitches and beyond: regulation of gene expression without proteins. *Nat. Rev. Genet.*, 8(10):776–790, 2007.

Shi, Y., Tyson, G. W., and DeLong, E. F. Metatranscriptomics reveals unique microbial small RNAs in the ocean's water column. *Nature*, 459(7244):266–269, 2009.

Siepel, A., Bejerano, G., Pedersen, J. S., Hinrichs, A. S., Hou, M., Rosenbloom, K., Clawson, H., Spieth, J., Hillier, L. W., Richards, S., Weinstock, G. M., Wilson, R. K., Gibbs, R. A., Kent, W. J., Miller, W., and Haussler, D. Evolutionarily conserved elements in vertebrate, insect, worm, and yeast genomes. *Genome Res.*, 15(8):1034–1050, 2005.

Smith, T. F. and Waterman, M. S. Identification of common molecular subsequences. *J. Mol. Biol.*, 147(1):195–197, 1981.

Smulevitch, S., Michalowski, D., Zolotukhin, A. S., Schneider, R., Bear, J., Roth, P., Pavlakis, G. N., and Felber, B. K. Structural and functional analysis of the RNA transport element, a member of an extensive family present in the mouse genome. *J. Virol.*, 79(4):2356–2365, 2005.

Sokal, R. R. and Michener, C. D. A statistical method for evaluating systematic relationships. *University of Kansas Science Bulletin*, 38:1409–1438, 1958.

Stadler, P. F., Chen, J. J., Hackermuller, J., Hoffmann, S., Horn, F., Khaitovich, P., Kretzschmar, A. K., Mosig, A., Prohaska, S. J., Qi, X., Schutt, K., and Ullmann, K. Evolution of vault RNAs. *Mol. Biol. Evol.*, 26(9):1975–1991, 2009.

Stocsits, R. R., Letsch, H., Hertel, J., Misof, B., and Stadler, P. F. Accurate and efficient reconstruction of deep phylogenies from structured RNAs. *Nucleic Acids Res.*, 37(18):6184–6193, 2009.

Taft, R. J., Pheasant, M., and Mattick, J. S. The relationship between non-protein-coding DNA and eukaryotic complexity. *Bioessays*, 29(3):288–299, 2007.

Taft, R. J., Glazov, E. A., Lassmann, T., Hayashizaki, Y., Carninci, P., and Mattick, J. S. Small RNAs derived from snoRNAs. *RNA*, 15(7):1233–1240, 2009a.

Taft, R. J., Kaplan, C. D., Simons, C., and Mattick, J. S. Evolution, biogenesis and function of promoter-associated RNAs. *Cell Cycle*, 8(15):2332–2338, 2009b.

Thompson, J. D., Higgins, D. G., and Gibson, T. J. CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. *Nucleic Acids Res.*, 22(22):4673–4680, 1994.

Torarinsson, E., Sawera, M., Havgaard, J. H., Fredholm, M., and Gorodkin, J. Thousands of corresponding human and mouse genomic regions unalignable in primary sequence contain common RNA structure. *Genome Res.*, 16(7):885–889, 2006.

Torarinsson, E., Havgaard, J. H., and Gorodkin, J. Multiple structural alignment and clustering of RNA sequences. *Bioinformatics*, 23(8):926–932, 2007.

Torarinsson, E., Yao, Z., Wiklund, E. D., Bramsen, J. B., Hansen, C., Kjems, J., Tommerup, N., Ruzzo, W. L., and Gorodkin, J. Comparative genomics beyond sequence-based alignments: RNA structures in the ENCODE regions. *Genome Res.*, 18(2):242–251, 2008.

Vapnik, V. N. *Statistical Learning Theory*. Wiley, New York, 1998.

Wang, A. X., Ruzzo, W. L., and Tompa, M. How accurately is ncRNA aligned within whole-genome multiple alignments? *BMC Bioinformatics*, 8:417, 2007.

Washietl, S., Hofacker, I. L., and Stadler, P. F. Fast and reliable prediction of noncoding RNAs. *Proc. Natl. Acad. Sci. U.S.A.*, 102(7):2454–2459, 2005.

Watts, J. M., Dang, K. K., Gorelick, R. J., Leonard, C. W., Bess, J. W., Swanstrom, R., Burch, C. L., and Weeks, K. M. Architecture and secondary structure of an entire HIV-1 RNA genome. *Nature*, 460(7256):711–716, 2009.

Weinberg, Z., Perreault, J., Meyer, M. M., and Breaker, R. R. Exceptional structured noncoding RNAs revealed by bacterial metagenome analysis. *Nature*, 462(7273):656–659, 2009.

Weinberg, Z., Wang, J. X., Bogue, J., Yang, J., Corbino, K., Moy, R. H., and Breaker, R. R. Comparative genomics reveals 104 candidate structured RNAs from bacteria, archaea, and their metagenomes. *Genome Biol.*, 11(3):R31, 2010.

Will, S., Reiche, K., Hofacker, I. L., Stadler, P. F., and Backofen, R. Inferring noncoding RNA families and classes by means of genome-scale structure-based clustering. *PLoS Comput. Biol.*, 3 (4):e65, 2007.

Wilm, A., Mainz, I., and Steger, G. An enhanced RNA alignment benchmark for sequence alignment programs. *Algorithms Mol. Biol.*, 1:19, 2006.

Zuker, M. and Stiegler, P. Optimal computer folding of large RNA sequences using thermodynamics and auxiliary information. *Nucleic Acids Res.*, 9(1):133–148, 1981.

Appendix A

List of publications

Journal papers (related to this dissertation)

1. Yutaka Saito, Kengo Sato, and Yasubumi Sakakibara, Robust and accurate prediction of noncoding RNAs from aligned sequences, *BMC Bioinformatics*, 11(Suppl 7):S3, 2010.
2. Yutaka Saito, Kengo Sato, and Yasubumi Sakakibara, Fast and accurate clustering of noncoding RNAs using ensembles of sequence alignments and secondary structures, *BMC Bioinformatics*, 12(Suppl 1):S48, 2011.

Journal papers (others)

1. Kensuke Morita, Yutaka Saito, Kengo Sato, Kotaro Oka, Kohji Hotta, and Yasubumi Sakakibara, Genome-wide searching with base-pairing kernel functions for noncoding RNAs: computational and expression analysis of snoRNA families in *Caenorhabditis elegans*, *Nucleic Acids Research*, 37(3):999–1009, 2009. (The first three authors are the joint Fisrt Authors.)
2. Yohei Okada, Yutaka Saito, Kengo Sato, and Yasubumi Sakakibara, Improved measurements of RNA structure conservation with generalized centroid estimators, *Frontiers in Genetics*, 2:54, 2011.

Conference proceedings (peer-reviewed full-length papers)

1. Kengo Sato, Yutaka Saito, and Yasubumi Sakakibara, Gradient-based optimization of hyperparameters for base-pairing profile local alignment kernels, *Proceedings of the 20th International Conference on Genome Informatics (GIW2009)*, pp. 128–138, Yokohama, Japan, Dec. 2009.

International conferences

1. Masaya Abe, Sumitaka Hase, Masahiro Ogawa, Yohei Okada, Kengo Sato, Yutaka Saito, and Yasubumi Sakakibara(*), Comprehensive analysis of small non-coding RNAs in medaka transcriptome by deep RNA-seq approach, *The 16th Annual Meeting of the RNA Society (RNA 2011)*, Kyoto, Japan, Jun. 2011. (oral presentation)
2. Yutaka Saito(*) and Yasubumi Sakakibara(*), Genome-wide detections of non-coding RNAs on genomes using kernel functions, *The International Workshop on Computational Methods for RNA analysis (Benasque 2009)*, Benasque, Spain, Aug. 2009. (oral presentation)
3. Kengo Sato(*), Yutaka Saito, and Yasubumi Sakakibara, Base-pairing profile local alignment kernels for functional RNA analyses, *The 17th Annual International Conference on Intelligent Systems for Molecular Biology and The 8th European Conference on Computational Biology (ISMB/ECCB 2009)*, Stockholm, Sweden, Jun. 2009. (poster presentation)