**Structural substitution matrices for RNA**

Marco Pietrosanto\*, Marta Adinolfi\*, Fabrizio Ferrè, Gabriele Ausiello and Manuela Helmer-Citterich

\*These authors contributed equally to the work

**ABSTRACT**

**INTRODUCTION**

Structural representation of RNA secondary (and tertiary) structure, is in constant evolution. The standard dot-bracket notation, although well-known and widely used, lacks of usability in terms of added information to any problem involving secondary structures. We must here divide the concept of secondary structures notation (or encoding) into two main focuses: the encoding can be used as a means to represent the final data (e.g. the secondary structure itself), in order to better visualize the problem. This is part of a communication effort which does not require the notation to be information rich in terms of “added features”. When a notation is used in this way, we call it *endpoint*. The notation can be used, however, to bring information to the problem addressed, in the same way “feature engineering” in the machine learning field greatly benefits the study. It can add useful aspects where they may be otherwise not exposed or simply not exploitable by the implemented approaches. When a notation is used in this way, we call it *featuring*. These two aspects are obviously not mutually exclusive but each notation is usually unbalanced towards one of the two characteristics.

As examples, the dot-bracket is used quite as an *endpoint* but is not a great *featuring* notation; any graph modeling of the secondary structure is an almost pure *featuring* notation, but lacks an immediate communication (1) and is usually algorithmically complex to use for large datasets. The BEAR notation, introduced by Mattei et al. (2), has proven useful to lower algorithm complexity for both structural pairwise alignments and large datasets motif discovery (3), yet lacks the powerful communication of other string-based structural representations such as the one used in other works (4–8), which are however, mostly used as *endpoints*.

The importance of a well-balanced notation is hence called for, and in this study we want to establish a framework that can be used by researchers to move in that direction. We present some possible string-based secondary structure encoding, inspired by some recent and/or well known work, but enriching the notation with substitution matrices which have proven useful for other tasks such as pairwise alignments and motif discovery. We show some possible applications that are directly derived by the possibility of using a *featuring* encoding and build some different structural groupings starting from RFAM families. An alternative structural conservation index is proposed, which is based on structural similarities of secondary structure elements and can shed light on strongly conserved structural motifs in multiple alignments of functionally related RNAs.

**MATERIALS AND METHODS**

* Curated RNA folding
* Structural Encodings
* RNA Blocks
* Beagle (tweaked for good with in-house script)
* - Hierarchical clustering
* Structural PSSM construction
* - Information Content

***Curated RNA folding***

The field of RNA folding prediction has been recently moving forward towards the usage of chemical probing outputs (9–11), with excellent results in predicting the secondary structure of RNAs (from an accuracy of ~70% to >90%) (12, 13). However, the experimental data is still not covering the massive amount of needed information to predict most of RNA molecules, and methods have been developed to meta-predict the chemical probing data itself (14). These methods are still at the dawn of their development and thus, many other applications still use standard folding predictors when in need of RNA structures (6, 15). To derive the secondary structures needed for this work we applied a method devised in (2),where RFAM seed members were each folded using hard constraints derived from the belonging consensus primary and secondary structure. In this way, an enhanced folding was obtained as described in the original paper.

***Structural Encodings***

Many different mappings can be drawn from structure contexts to string representation and the BEAR encoding shown that efficiently writing a 2D structure into a linear string can be useful to bring down algorithm complexity, especially for large scale data (3, 16). Recently, another encoding has used typically complex structures such as graphs to describe secondary structures (and pseudoknots) efficiently (8), and one of the endpoints of the algorithm is a 8-character string describing structural contexts. Enhanced alphabets have been used to describe secondary structures as endpoints of other applications, such as *in vitro* short motif discovery (5, 17) and [REF ALTRA ROBA CHE HANNO TIRATO FUORI CONTEXT ENCODING OLTRE AL CANADESE INFAME PER TE SOLO LAME]. The importance of the encoding choice is dual: on one hand it may be crucial if it is used mid-algorithm as a means to do “feature engineering”, that is letting important aspects emerge from the studied subjects in such a way a downhill algorithm can benefit of this pre-processing; on the other hand an optimal string encoding is better understood if it is readable, easily interpretable and conveys any contained message efficiently. The standard BEAR encoding works well in terms of usability and performance (see original paper, (2)) but lacks communication. The other string representations used instead, lack of usability as they are mostly used as endpoints (extremizing the communication part, but without any added feature value). In this study we wanted to scale down the intricacy of the standard BEAR, while generalizing the framework of structural substitution matrices and their applications.

We re-worked the BEAR substitution matrix, introduced two new proposed alphabets (of increasing interpretability) and built their substitution matrices.

### FORSE FINO A QUA E’ PIU’ INTRODUZIONE

The qBEAR (quickBEAR) has appeared in the original BEAM paper (3) as a means of representing the logo of a secondary structure motif. It divides the structural contexts in 3 groups each based on the distribution of context lengths (see Supplementary Data), ending with an alphabet composed of 18 characters.

The zBEAR (zipBEAR) is instead inspired by the few-characters alphabet mostly used as end-points (6, 8), where each different context is described by one character, ending up with an alphabet composed of 8 characters.

Necessità per la questione della lunghezza, la scarsa comprensibilità, il fatto che un sacco di gente pubblica con delle codifiche minimali e che noi dobbiamo un po’ gestire i differenti elementi altrimenti le matrici di sostituzione diventano inutili. Si vede infatti che bear originale è comunque il migliore per allineare, ma il prezzo da pagare è la scarsa comprensibilità immediata. Qbear è uscito con il lavoro di beam come mezzo per visualizzare meglio i loghi strutturali, Zbear è una compressione ulteriore del bear (zipBear) che imita la larghezza della maggior parte degli encoding utilizzati in altri lavori, nell’ottica di avere un alfabeto comprensibile anche a colpo d’occhio.

***RNA Blocks***

To build a framework from where different substitution matrices could be derived, we were inspired by the classical formulation of BLOSUM BLOCKS, with some relaxations (18, 19). For each RFAM seed alignment, after a redundancy removal (which becomes a parameter in the construction of the matrix, in this work we used 62% and 90%) a column was chosen to be part of an RNA Block if 1. no gaps where present and 2. a structural consensus, dependent on the chosen alphabet, must exist (i.e. there must be a structural character with a relative frequency >50%).

Each column is then included in the Block of that family. The relaxation here is the non-necessity to have contiguous columns to form a block, since we believe that single column information is sufficient to derive a structural substitution matrix.

For each encoding we thus have a set of blocks to derive the substitution matrix, as per (2, 20).

###IMMAGINI DEL PROCESSO DEI BLOCCHI E DELLE MATRICIZZAZIONI

***Beagle***

The recently developed Beagle (16) was used to compute pairwise alignments of benchmarks and between RNA consensus. This method applies a custom dynamic programming algorithm to make pairwise alignments between structures, with the aid of a sequence bonus. For each encoding we determine the best sequence bonus based on the SPS performance of 4 benchmarks datasets (Supplementary Data).

The benchmarks are the same used in the original Beagle paper (Bralibase II, RNAstrand, RNAspa, RRS) (21–23). For each matrix, a set of different bonuses were tested in the range of [0,10] and SPS was used as a performance measure, with the sequence bonus granting the best SPS used as a standard for each matrix. The SPS formula is reported here:

where is the aligned pair in the computed alignment and is the aligned pair in the reference alignment. The is 1 if and only if and is the length of the reference alignment. In the rare case the two alignments are of different length, the longest one is tested on every possible window where the smallest alignment is completely contained in the longest one, taking the best SPS among the possible ones.

The RNA consensus (sequence and structures) were instead used to build an RFAM structural tree, in particular we derived a NxN matrix from the similarity scores between each pair of consensus (where N is the number of RFAM families with both a consensus sequence and structure). This was transformed into a triangular distance matrix with the following mapping:

####### FORMULA

At last, a hierarchical clustering algorithm was applied to derive the tree, and ---- COSE CHE DECIDEREMO DI FARE CON L’ALBERO

***Structural PSSM construction***

A given encoding may be used to build models from alignments. We started off with PSSMs (Position Specific Scoring Matrices) as defined in the original formulation of Eisenberg (24). In particular, given each RFAM family and its seed alignment, we are able to build a structural PSSM model of that family, depending on the chosen encoding and its relative substitution matrix. A single PSSM is built in the following way:

Where is the position index and runs over all the alphabet’s characters

***Information Content and Relative Information Gain***

Each structural PSSM contains information about the conservation of certain structural contexts. To extract this feature we use the Shannon Entropy, which can be seen as the amount of extra-bits (if a log in base-2 is used in the formula) needed to describe the distribution of a given PSSM column:

where *p* is a probability distribution. This is 0 when the distribution is completely unimodal (i.e. a single character is present in the column, no extra-bits needed to completely understand the distribution), and is maximum when the distribution is uniform (i.e. we need to specify every single character with one bit each). The maximum is , where is the number of available characters and, in this form, is dependent on the encoding chosen.

However, since a PSSM column is not normalized as a probability distribution, we apply a transformation which first linearizes the quantities by applying the exponential function (a PSSM cell is proportional to a log-odd) and then renormalize the values such that the sum of the values in a column is 1. In this way the ordering between values is preserved and the added value of a PSSM, which is the information contained in the substitution matrix, is brought forth in the probability vector.

Since we are developing a framework that must work cross-encodings, and we want a measure of relative structural conservation of the column, we derived the following quantity:

This ensures 1. that the measure is normalized between 0 and 1 for every possible encoding and 2. it has the intuitive property of being 1 when the structure is conserved in the column in each of the alignment members and 0 when the structure is not conserved at all.

**RESULTS AND DISCUSSION**

* Substitution Matrices
* Performance on pairwise alignments
* Structural Trees
* RFAM PSSMs as family models
* RFAM classification ?? (poor performances, can be explained with structural trees)
* Relative Information Gain (RIG)
* Discussion
  + Different substitution matrices for different usages
    - Pippone on the alphabet choice
  + They help extract information in a structural oriented way
    - RIG as indication of structurally conserved sections
    - Trees to let similar structures emerge across families (as orthogonal information to other groupings, such as clans)
  + La classificazione sucks, but other works (such as ProtVec) demonstrated that an efficient representation (such as NN embeddings) can be used in pair with simple methods (SVM) to reach astonishing classification results, thus we welcome to try and move in that direction.

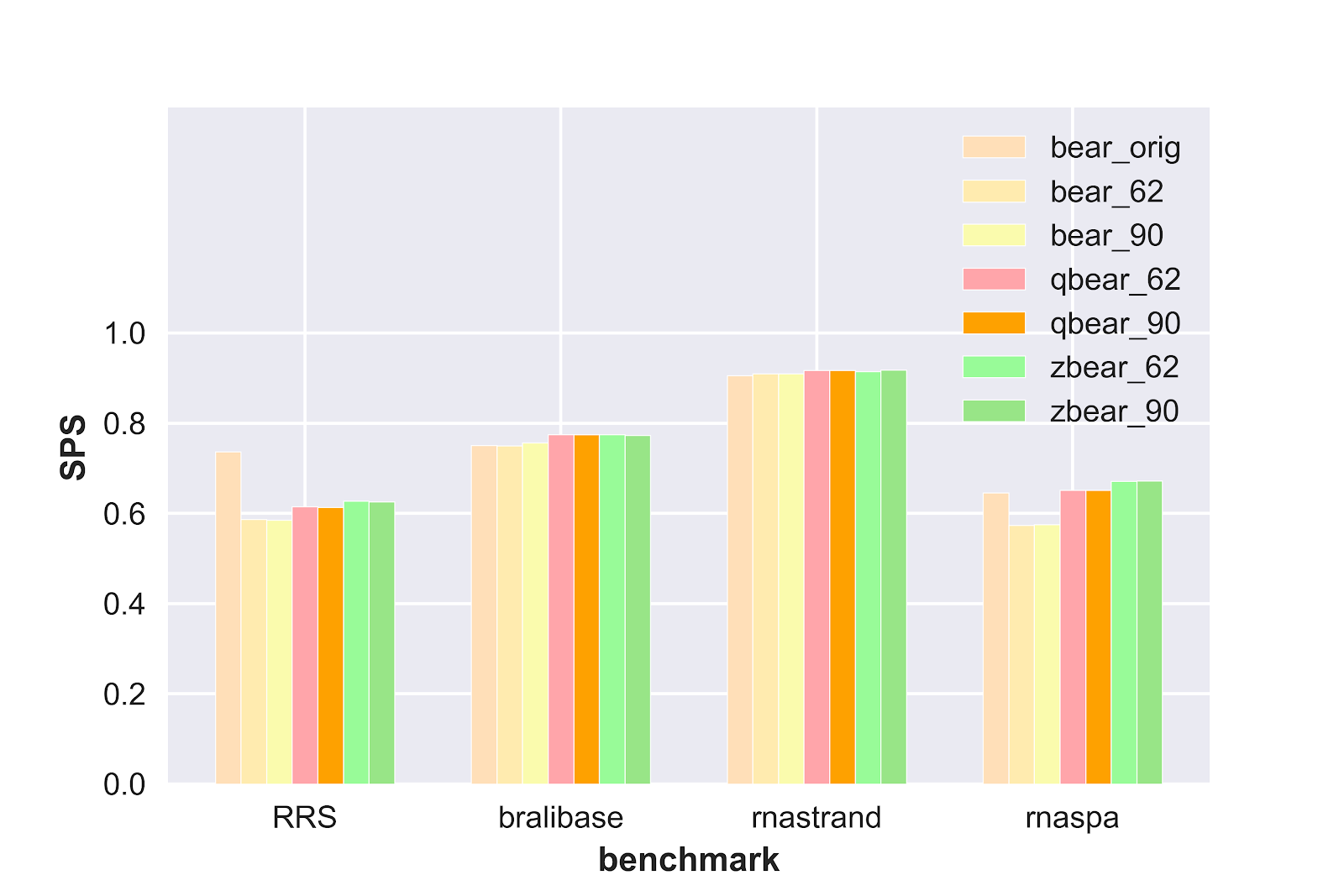
***Substitution Matrices***

With the resulting framework it is possible to build a substitution matrix off of any redundancy level and encoding chosen. For this specific study, we have chosen to work with a rebuilt BEAR matrix (83x83), a qBEAR matrix (18x18) and a zBEAR matrix (8x8), each with two different levels of redundancy removal: 62% and 90%. In particular we expect to see a trade-off between immediacy of the encoding used and the amount of information we are able to retrieve with different models. Rich encodings like BEAR is expected to work better in fine-grained tasks, as is aligning two sequences (since it involves summation of many terms and differences can be defined by a single character we expect a complex encoding to be more functional). Simpler encodings like qBEAR and zBEAR instead should be useful to catch general properties of the data, like distribution dependent measures (e.g. Information Content and Structural neighboring) while at the same being readable and easily interpretable in a visual context.

../rna-division/beagleUser/plots/Zbear90.pdf../rna-division/beagleUser/plots/qbear62.pdf../rna-division/beagleUser/plots/bear90.pdf

***Performance on pairwise alignments***

The performances of the newly created matrices are [alike?] the previous results of the original Beagle work (2), with the new version of the BEAR matrix having a slightly worse performance than the original in terms of sequence SPS, this is probably due to the amount of heterogeneous data used (XX families against the old .. 17?, probably enriching the structural information at the expense of the sequence). Yet we strongly believe that a well-defined framework will bring benefits in the long run. The other matrices, as expected, have lower performances for two of the four datasets tested with respect to the standard BEAR, but still they provide correct alignments.



***Structural Trees***

Pairwise alignments however, can be efficiently used to create groupings with respect to features of interest. In particular, by aligning each RFAM family consensus sequence and structure with Beagle, we derive a triangular distance matrix, as described in the Methods section. The hierachical clustering reveals a new grouping of RNA families, independent from a purely functional one (i.e. the clans provided by RFAM, FIGURA DOVE I CLAN SONO SEPARATI).

By cutting the tree at a distance which is the nearest multiple of 5\*10^-2 to the median of the distance matrix, we can see that the groups are extremely alike in terms of structure, bringing the field an unapproached point of view. For example ####ESEMPIO DI FAMIGLIE CHE NON T’ASPETTI MA CHE STANNO VICINE

###IMMAGINI DI QUESTA COSA

***RFAM PSSMs as family models***

***RFAM classification ?? (poor performances, can be explained with structural trees)***

***Relative Information Gain (RIG)***

An alternative structural conservation index is proposed, which is based on structural similarities of secondary structure elements and can shed light on strongly conserved structural motifs in multiple alignments of functionally related RNAs.

../pssm_proj/results/RIG/qbear_90/qbear_90_RF02021_delogit2.pdf

Figure 1. qBEAR90 RIG of RF02021

../pssm_proj/results/RIG/qbear_90/qbear_90_RF02264_delogit2.pdf

Figure 2. qBEAR90 RIG of RF02264 … interpretation

../pssm_proj/results/RIG/qbear_90/qbear_90_RF02230_delogit2.pdf

Figure 3. qBEAR90 RIG of RF02230 … interpretation palazzi (magari ci mettiamo accanto la struttura cons?)

***Discussion***

* + Different substitution matrices for different usages
    - Pippone on the alphabet choice
  + They help extract information in a structural oriented way
    - RIG as indication of structurally conserved sections
    - Trees to let similar structures emerge across families (as orthogonal information to other groupings, such as clans)
  + La classificazione sucks, but other works (such as ProtVec) demonstrated that an efficient representation (such as NN embeddings) can be used in pair with simple methods (SVM) to reach astonishing classification results, thus we welcome to try and move in that direction.

**SUPPLEMENTARY DATA**

**ACKNOWLEDGEMENT**

**FUNDING**

**REFERENCES**

1. Maticzka,D., Lange,S.J., Costa,F. and Backofen,R. (2014) GraphProt: modeling binding preferences of RNA-binding proteins. *Genome Biol.*, **15**, R17.

2. Mattei,E., Ausiello,G., Ferrè,F. and Helmer-Citterich,M. (2014) A novel approach to represent and compare RNA secondary structures. *Nucleic Acids Res.*, **42**, 6146–6157.

3. Pietrosanto,M., Mattei,E., Helmer-Citterich,M. and Ferrè,F. (2016) A novel method for the identification of conserved structural patterns in RNA: From small scale to high-throughput applications. *Nucleic Acids Res.*, **44**, 8600–8609.

4. Li,X., Quon,G., Lipshitz,H.D. and Morris,Q. (2010) Predicting in vivo binding sites of RNA-binding proteins using mRNA secondary structure. *RNA*, **16**, 1096–107.

5. Cook,K.B., Hughes,T.R. and Morris,Q.D. (2015) High-throughput characterization of protein-RNA interactions. *Brief. Funct. Genomics*, **14**, 74–89.

6. Polishchuk,M., Paz,I., Kohen,R., Mesika,R., Yakhini,Z. and Mandel-Gutfreund,Y. (2017) A combined sequence and structure based method for discovering enriched motifs in RNA from in vivo binding data. *Methods*, **118**–**119**, 73–81.

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

8. Danaee,P., Rouches,M., Wiley,M., Deng,D., Huang,L. and Hendrix,D. (2018) bpRNA: large-scale automated annotation and analysis of RNA secondary structure. *Nucleic Acids Res.*, **46**, 1–14.

9. Kertesz,M., Wan,Y., Mazor,E., Rinn,J.L., Nutter,R.C., Chang,H.Y. and Segal,E. (2010) Genome-wide measurement of RNA secondary structure in yeast. *Nature*, **467**, 103–107.

10. Wan,Y., Qu,K., Ouyang,Z. and Chang,H.Y. (2013) Genome-wide mapping of RNA structure using nuclease digestion and high-throughput sequencing. *Nat. Protoc.*, **8**, 849–869.

11. Wan,Y., Qu,K., Zhang,Q.C., Flynn,R. a, Manor,O., Ouyang,Z., Zhang,J., Spitale,R.C., Snyder,M.P., Segal,E., *et al.* (2014) Landscape and variation of RNA secondary structure across the human transcriptome. *Nature*, **505**, 706–9.

12. Washietl,S., Hofacker,I.L., Stadler,P.F. and Kellis,M. (2012) RNA folding with soft constraints: Reconciliation of probing data and thermodynamic secondary structure prediction. *Nucleic Acids Res.*, **40**, 4261–4272.

13. Lorenz,R., Hofacker,I.L. and Stadler,P.F. (2016) RNA folding with hard and soft constraints. *Algorithms Mol. Biol.*, **11**, 8.

14. Delli Ponti,R., Marti,S., Armaos,A. and Tartaglia,G.G. (2017) A high-throughput approach to profile RNA structure. *Nucleic Acids Res.*, **45**, e35–e35.

15. Navarin,N. and Costa,F. (2017) An efficient graph kernel method for non-coding RNA functional prediction. *Bioinformatics*, **33**, 2642–2650.

16. Mattei,E., Pietrosanto,M., Ferr??,F. and Helmer-Citterich,M. (2015) Web-Beagle: A web server for the alignment of RNA secondary structures. *Nucleic Acids Res.*, **43**, W493–W497.

17. Kazan,H. and Morris,Q. (2013) RBPmotif: a web server for the discovery of sequence and structure preferences of RNA-binding proteins. *Nucleic Acids Res.* , **41**, W180–W186.

18. Henikoff,S., Henikoff,J.G. and Pietrokovski,S. (1999) Blocks+: a non-redundant database of protein alignment blocks derived from multiple compilations. *Bioinformatics*, **15**, 471–479.

19. Pietrokovski,S., Henikoff,J.G. and Henikoff,S. (1996) The Blocks Database—A System for Protein Classification. *Nucleic Acids Res.*, **24**, 197–200.

20. Henikoff,S. and Henikoff,J.G. (1992) Amino acid substitution matrices from protein blocks. *Proc. Natl. Acad. Sci.*, **89**, 10915–10919.

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

22. Andronescu,M., Bereg,V., Hoos,H.H. and Condon,A. (2008) RNA STRAND: The RNA secondary structure and statistical analysis database. *BMC Bioinformatics*, **9**, 1–10.

23. Kalvari,I., Argasinska,J., Quinones-Olvera,N., Nawrocki,E.P., Rivas,E., Eddy,S.R., Bateman,A., Finn,R.D. and Petrov,A.I. (2017) Rfam 13.0: shifting to a genome-centric resource for non-coding RNA families. *Nucleic Acids Res.*, **46**, 335–342.

24. Gribskov,M., McLachlan,A. and Eisenberg,D. (1987) Profile analysis: detection of distantly related proteins. *Proc. Natl. Acad. Sci. U. S. A.*, **84**, 4355–4358.