Beyond Mfold with probabilistic models

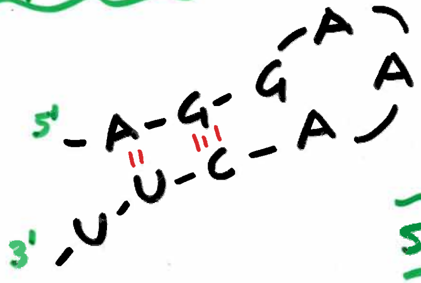
Tornado

a language for generating a large spectrum of complex context-free grammars for RNA secondary structure

A brief unifying description of RNA structure prediction

Models of RNA folding

Thermodynamic versus Statistical



Statistical

$$S \to \underset{t_1}{aS} \mid \underset{t_2}{Sa} \mid \underset{t_3}{aSa'} \mid \underset{t_4}{\varepsilon}$$

Thermo

	ΔG (kcal/mol)
U dangle A:U	-0.16
G:C stacked A:U	-3.41
hairpinloop 4 nts	+9.09
GAAA bonus	- 3.97
Total Free Energy :	+1.55 kcal/mol

	S		
	↓ S	U	$\log t_2 + \log P(U)$
A	↓ S	U	$\log t_3 + \log P(A\ U)$
G	↓ S	C	$\log t_3 + \log P(G\ C)$
G	↓ S		$\log t_1 + \log P(G)$
A	↓ S		$\log t_1 + \log P(A)$
A	↓ S		$\log t_1 + \log P(A)$
A	↓ S		$\log t_1 + \log P(A)$
ε	↓ S		$\log t_4$

$\log P(\text{total}) = \Sigma$ all terms

Free energies ~ log P's

Most thermo models admit a grammar description

Same algorithms, $O(L^3)$ in time

Going beyond thermodynamic models

One very **complicated** thermodynamic model to several extremely **simple** probabilistic models

Thermodynamic models outperform Probabilistic models

Grammar	Parameters	best F % (by posterior decoding)	
G6	21	48	probabilistic
ViennaRNA	∼14,000	54	Thermodynamic

Still performance is poor

Believe: probabilistic models are **too constrained and cannot** implement all the complexities of the thermodynamic models. Need to move to other type of statistical methods.

Why Statistical Models?

specifically with probabilistic parameters

Statistical models learn parameters from known RNA structures which is an **ever-growing** source of information versus the **slowly-produced** thermodynamic parameters.

Statistical **non-probabilistic** models:

CONTRAfold	Do, Woods, Batzoglou ’06
Simfold	Andronescu *et al.* ‘07 & ‘10

Advantage of statistical **probabilistic** models:

Easily Trainable Can train on large corpus of data

Generative can interrogate the model by sampling
can rationally change properties of the model
(target length or target base composition)

Optimal comparison of alternative hypotheses

(Newman & Pearson ’33)

Easy integration of complementary sources of information

TORNADO

A compact description of RNA grammars

is a big fat general RNA model that can accomodate most element of RNA 2D structure and beyond one could think of.

flexible: Fast model exploration / Probabilistic or not
robust: One folding algorithm for all models

tool to be able to test many different models

A “basic” complex grammar

S	$\longrightarrow$	$\mathrm{a}\ \ \mathrm{S}\ \mid\ \mathrm{F0}\ \ \mathrm{S}\ \mid\ \epsilon$	
F0	$\longrightarrow$	$\mathrm{a}\ \ \mathrm{F5}\ \ \mathrm{a}'$	#Helix starts
F0	$\longrightarrow$	$\mathrm{a}\ \ \mathrm{P}\ \ \mathrm{a}'$	#Helix (of 1 pair) ends
F5	$\longrightarrow$	$\mathrm{a}\ \ \mathrm{F5}\ \ \mathrm{a}'$	# Helix continues
F5	$\longrightarrow$	$\mathrm{a}\ \ \mathrm{P}\ \ \mathrm{a}'$	# Helix ends
P	$\longrightarrow$	$\mathrm{a}_1 ... \mathrm{a}_n$	# hairpin loop
P	$\longrightarrow$	$\mathrm{a}_1 ... \mathrm{a}_n\ \ \mathrm{F0}$	# left-bulges
P	$\longrightarrow$	$\mathrm{F0}\ \ \mathrm{a}_1 ... \mathrm{a}_n$	# right-bulges
P	$\longrightarrow$	$\mathrm{a}_1 ... \mathrm{a}_n\ \ \mathrm{F0}\ \ \mathrm{a}_{n+1} ... \mathrm{a}_m$	# internal loops
P	$\longrightarrow$	$\mathrm{M1}\ \ \mathrm{M}$	# multiloop (TWO or more helices)
M1	$\longrightarrow$	$\mathrm{a}\ \ \mathrm{M1}\ \mid\ \mathrm{F0}$	# ONE helix with bases to the left
M	$\longrightarrow$	$\mathrm{M1}\ \ \mathrm{M}\ \mid\ \mathrm{R}$	# ONE or more helices
R	$\longrightarrow$	$\mathrm{R}\ \ \mathrm{a}\ \mid\ \mathrm{M1}$	# last right helix

TORNADO language

basic_grammar

```
# BASIC GRAMMAR [Includes loops and stacking but no dangles]

# PARAMETER DEFINITIONS
# def : param name : param value
def : p-FIT_LENGTH : 30
def : p-MAX_LENGTH : p-FIT_LENGTH

# TRANSITION DISTRIBUTIONS
# tdist : n : t-name
tdist : 5 : t-P
tie : 1 : 2 # tie left and right bulges

# EMISSION DISTRIBUTIONS
# edist : nemit : ncontext : nbasepairs : basepair type : e-name
edist : 1 : 0 : 0 :        e1 # one single residue emission distribution
edist : 2 : 0 : 1 : _WW_ : e1 # one WW basepair distribution (helix opening)
edist : 2 : 0 : 1 : _WW_ : e2 # one WW basepair distribution (helix opening and closing)
edist : 2 : 2 : 1 : _WW_ : e1 # 16 WW basepair stacked distributions (helix extend)
edist : 2 : 2 : 1 : _WW_ : e2 # 16 WW basepair stacked distributions (helix closing)

# LENGTH DISTRIBUTIONS
# ldist : min : fit : max : l-name
# ldist-di : minL : minR : min sum : fit : max : l-name
ldist :             3 : p-FIT_LENGTH : p-MAX_LENGTH : l1 # Hairpin Loops
ldist :             1 : p-FIT_LENGTH : p-MAX_LENGTH : l2 # Bulges
ldist-di : 1 : 1 : 2 : p-FIT_LENGTH : p-MAX_LENGTH : l3 # Internal Loops

# RULES

 S  ->   a : i e1 S(i+1,j) | F0 S | e        # Start: a left base, or a left Helix, or End

F0  ->   a : i & j e1 F5(i+1,j-1)            # Helix starts
F0  ->   a : i & j e2 P (i+1,j-1)            # Helix (of one basepair) ends

F5  ->   a : i & j : i-1,j+1 e1 F5(i+1,j-1)  # Helix continues
F5  ->   a : i & j : i-1,j+1 e2 P (i+1,j-1)  # Helix ends

 P  ->   t-P  m...m(i,j) l1                              # Hairpin Loop
 P  ->   t-P  m...m(i,k) l2 F0(k+1,j)                    # Left Bulges
 P  ->   t-P               F0(i,k-1) m...m(k,j) l2       # Right Bulges
 P  ->   t-P  d...(i,k) ...d(l,j) l3 F0(k+1,l-1)         # Internal Loops
 P  ->   t-P  M2                                         # Multiloop

M2  ->   M1 M                        # TWO or more Helices
 M  ->   M1 M | R                    # ONE or more Helices
M1  ->   F0 | a : i e1 M1(i+1,j)     # ONE Helix, possibly with single left bases
 R  ->   M1 | R(i,j-1) a : j e1      # last Helix, possibly with left/right bases
```

Tornado features

Arbitrary residue emissions: Emissions can include an arbitrary number of residues, and can depend on an arbitrary number of previously emitted residues (contexts).

Stacked basepairs [P$^{c,\hat{c}}$ -> a F $\hat{a}$]:

In TORNADO language: a:i&j:i-1,j+1 F(i+1,j-1).

Hairpin mismatches [P$^{c,\hat{c}}$ -> a [m...m] b]:

In TORNADO language: a:i,j:i-1,j+1 m...m(i+1,j-1).

Tetraloops depending on closing basepair [P$^{c,\hat{c}}$ -> $a_1\,a_2\,a_3\,a_4$]:

In TORNADO language: a:i,i+1,i+2,i+3:i-1,j+1.

Internal loop mismatches [P$^{c,\hat{c}}$ -> a[d...]b F $\hat{b}$[...d]e]

In TORNADO language: a:i,j:i-1,j+1 d...(i+1,k)...d(l,j-1) F(k+2,l-2)

more TORNADO emissions

Other first order emissions tested with TORNADO, and not included in the standard NN model are:

dangles in bulges [`P`$^{c,\hat{c}}$ `->` a`[m...m]`b `F` $\hat{b}$]:

In TORNADO language: `a:i:i-1,j+1 m...m(i+1,k) b:k+1&j:k F(k+2,j-1)`.

mismatches (or dangles) in multiloops

unambiguously

coaxial stacking [`P -> `a` F `$\hat{a}$` `b` F `$\hat{b}$]:

In TORNADO language: `a:i&k b:j&k+1:i,k F(i+1,k-1) F(k+2,j-1)` or `a:i&k,j&k+1 F(i+1,k-1) F(k+2,j-1)`.

and more...

Tornado can also be used to build second (or higher) order Markov dependencies, rather than just first order. Examples are

- **dangles (or more than one single base)** depending on several bases [$\mathrm{P}^{c,d,e}$ `->` a F | a b F]:

  In Tornado language: `a:i:i-1,i-2,i-3 F(i+1,j)` and `a:i,i+1:i-1,i-2,i-3 F(i+2,j)`.

- **higher order stacked pairs** [$\mathrm{P}^{b,\hat{b},c,\hat{c}}$ `->` a F$\hat{a}$]:

  In Tornado language: `a:i&j:i-1,i-2,j+1,j+2 F(i+1,j-1)`.

- **three single bases depending on two basepairs** [$\mathrm{P}^{e,\hat{e},f,\hat{f}}$ `->` a b c F]:

  In Tornado language: `a:i,i+1,i+2:i-1,i-2,j+1,j+2 F(i+3,j)`.

other TORNADO features

Length distributions for loop emission: Mono-segment loops (for instance for hairpins, bulges, multiloops or external bases), and di-segment loops (for internal loops) can be specified.

Length distribution tails for loop emissions:

Length distributions for stems:

Arbitrary 4x4 canonical basepairs and non-canonical TORNADO allows distinguishing 12 types of basepairs

Specific values: These values could be free-energy changes obtained from thermodynamic data or arbitrary scores provided by other means.

Tying of parameters: to reuse emission and transition distribution and avoid a explosion of parameters.

tertiary interactions

```
# enhanced nussinov
# (an extension of grammar G5 to tertiary contacts)
#
# C. Honer zu Siederdissen and S. H. Bernhart, and P. F. Stadler and I. Hofacker
# "A folding algorithm for extended RNA secondary structures" Bioinformatics 27, i129-i136, 2011.

# singlet emission
edist : 1 : 0 : 0 : e1

# basepair emissions
edist : 2 : 0 : 1 : _WW_ : e1  # for no-triplet basepairs (e(i,j)   in paper)
edist : 2 : 0 : 1 : _WW_ : e2  # for left       triplets  (e^a(i,j) in paper)
edist : 2 : 0 : 1 : _WW_ : e3  # for right      triplets  (e^b(i,j) in paper)
edist : 2 : 0 : 1 : _WW_ : e4  # for left/right triplets  (e^c(i,j) in paper)

F  --> a:i   e1 F (i+1,j)   | a:i   e1
F  --> a:i&j e1 F (i+1,j-1) | a:i&k e1 F (i+1,k-1) F(k+1,j) # recursion for C can be spared
F  --> a:i&j e2 U1(i,  j-1) | a:i&k e2 U1(i,  k-1) F(k+1,j)
F  --> a:i&j e3 V (i+1,j)   | a:i&k e3 V (i+1,k)   F(k+1,j) | a:i&k e3 F (i+1,k-1) U1(k,j)
F  --> a:i&j e4 W1(i,  j)   | a:i&k e4 W1(i,  k)   F(k+1,j) | a:i&k e4 U1(i,  k-1) U1(k,j)

# left base of U1 has to basepair
U1 --> a:i&j e1 F(i+1,j-1)  | a:i&k e1 F(i+1,k-1) F (k+1,j)
U1 --> a:i&j e3 V(i+1,j)    | a:i&k e3 V(i+1,k)   F (k+1,j)
U1 -->                        a:i&k e4 F(i+1,k-1) U1(k,  j)

# right base of V has to basepair
V -->  a:i   e1 V (i+1,j)
V -->  a:i&j e1 F (i+1,j-1) | a:i&k e1 F (i+1,k-1) V(k+1,j)
V -->  a:i&j e2 U1(i,  j-1) | a:i&k e2 U1(i,  k-1) V(k+1,j) | a:i&k e2 U1(i,  k-1) W(k,j)
V -->                         a:i&k e3 V (i+1,k)   V(k+1,j) | a:i&k e3 F (i+1,k-1) W(k,j)
V -->                         a:i&k e4 W1(i,  k)   V(k+1,j)

#left and right bases of W have to basepair
W  --> a:i&j e4 F(i+1,j-1) | W1(i,j)

#left and right bases of W1 have to basepair but not to each other
W1 --> a:i&k e2 U1(i,  k-1) V(k+1,j)
W1 --> a:i&k e3 V (i+1,k)   V(k+1,j)
W1 --> a:i&k e4 F (i+1,k-1) W(k,  j)
```

Existing complex grammars

I have created TORNADO “emulations” of the state of the art RNA models that exist to date.

ViennaRNA
thermodynamic

ViennaRNA-G
TORNADO grammar

14,000 parameters

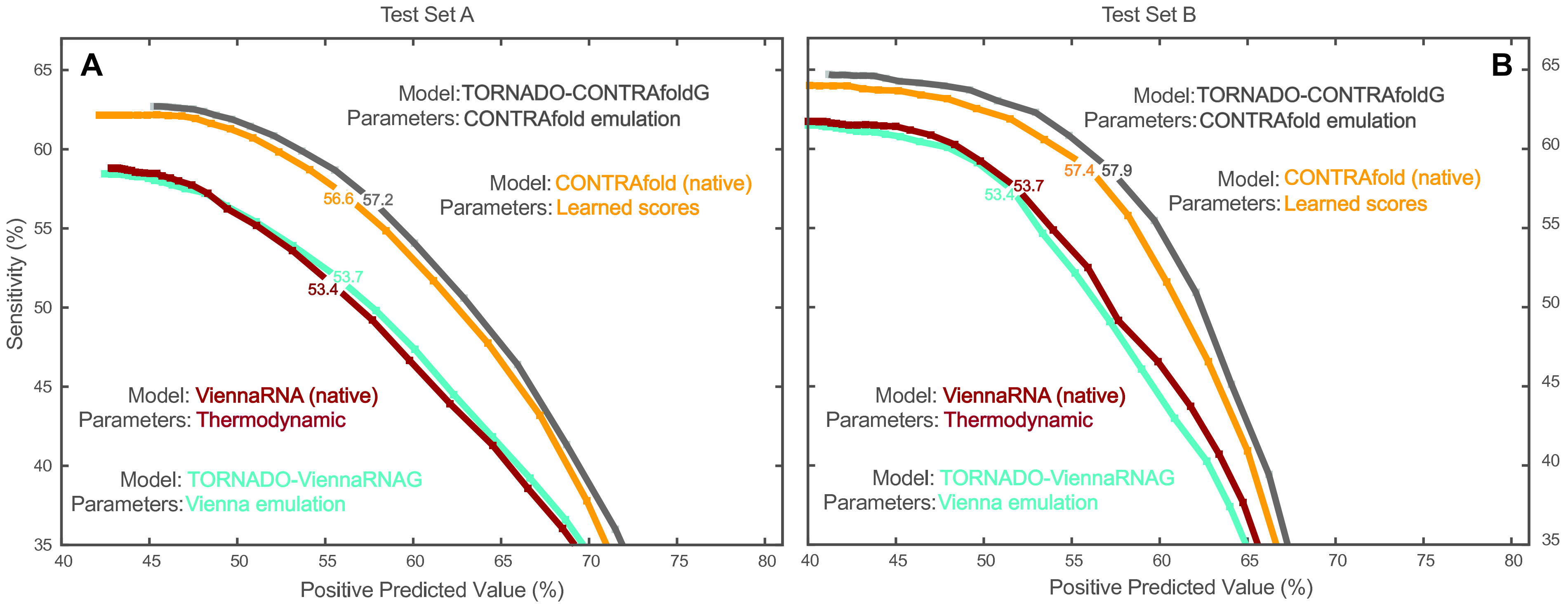
ContraFOLD
learned parameters

ContraFOLD-G
TORNADO grammar

1,500 parameters

TORNADO-emulations

We have probabilistic models that reproduce the complexity of the thermodynamic nearest-neighbor model.



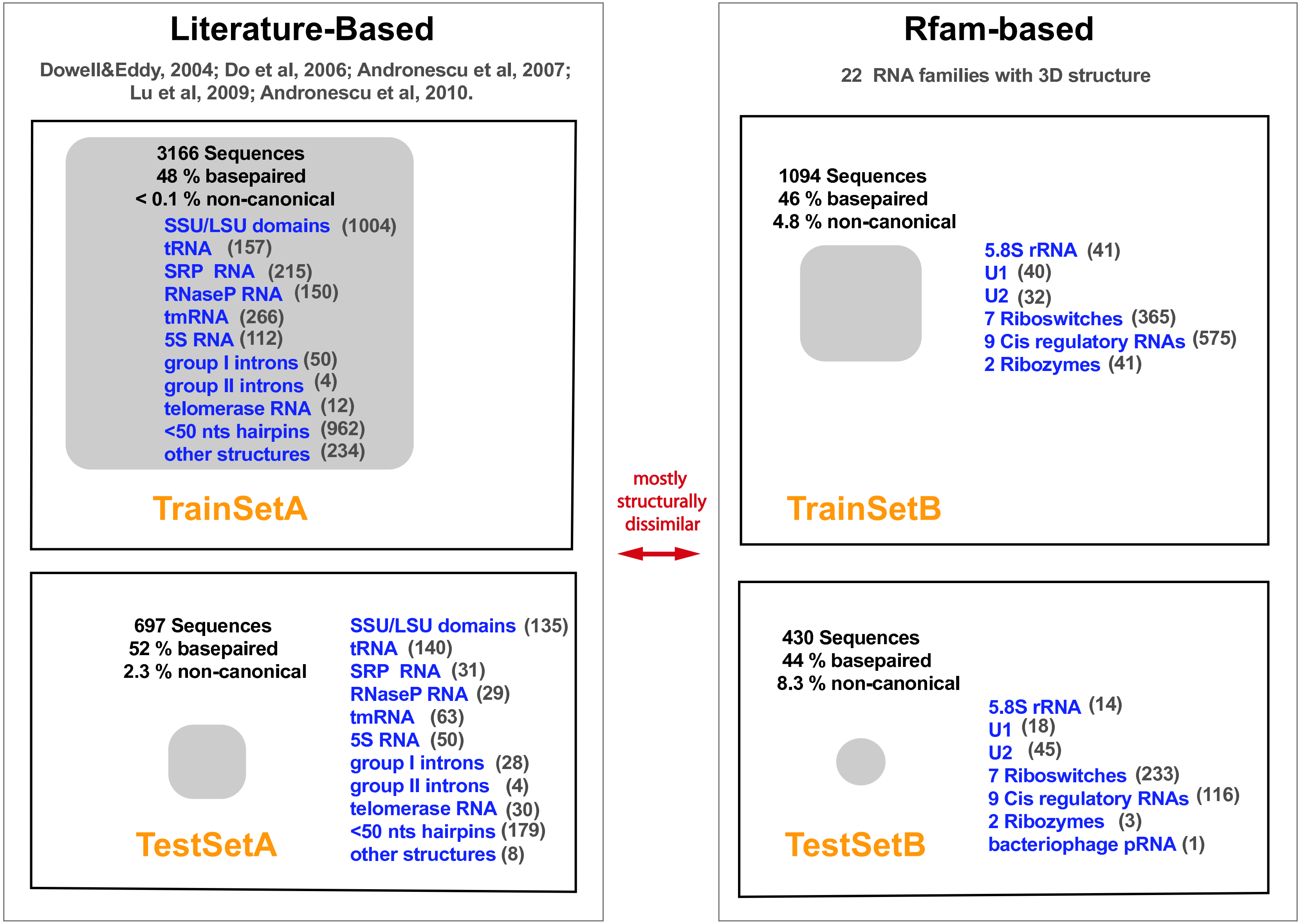
ViennaRNA and CONTRAfold

Probabilistic Complex Grammars

What happens if now one turns the parameters of these models into **probabilities** trained using known RNA structures?

Benchmark tools

Training and test sets



Literature-Based

Dowell&Eddy, 2004; Do et al, 2006; Andronescu et al, 2007;
Lu et al, 2009; Andronescu et al, 2010.

TrainSetA

3166 Sequences
48 % basepaired
< 0.1 % non-canonical

SSU/LSU domains	(1004)
tRNA	(157)
SRP RNA	(215)
RNaseP RNA	(150)
tmRNA	(266)
5S RNA	(112)
group I introns	(50)
group II introns	(4)
telomerase RNA	(12)
<50 nts hairpins	(962)
other structures	(234)

TestSetA

697 Sequences
52 % basepaired
2.3 % non-canonical

SSU/LSU domains	(135)
tRNA	(140)
SRP RNA	(31)
RNaseP RNA	(29)
tmRNA	(63)
5S RNA	(50)
group I introns	(28)
group II introns	(4)
telomerase RNA	(30)
<50 nts hairpins	(179)
other structures	(8)

mostly
structurally
dissimilar

Rfam-based

22 RNA families with 3D structure

TrainSetB

1094 Sequences
46 % basepaired
4.8 % non-canonical

5.8S rRNA	(41)
U1	(40)
U2	(32)
7 Riboswitches	(365)
9 Cis regulatory RNAs	(575)
2 Ribozymes	(41)

TestSetB

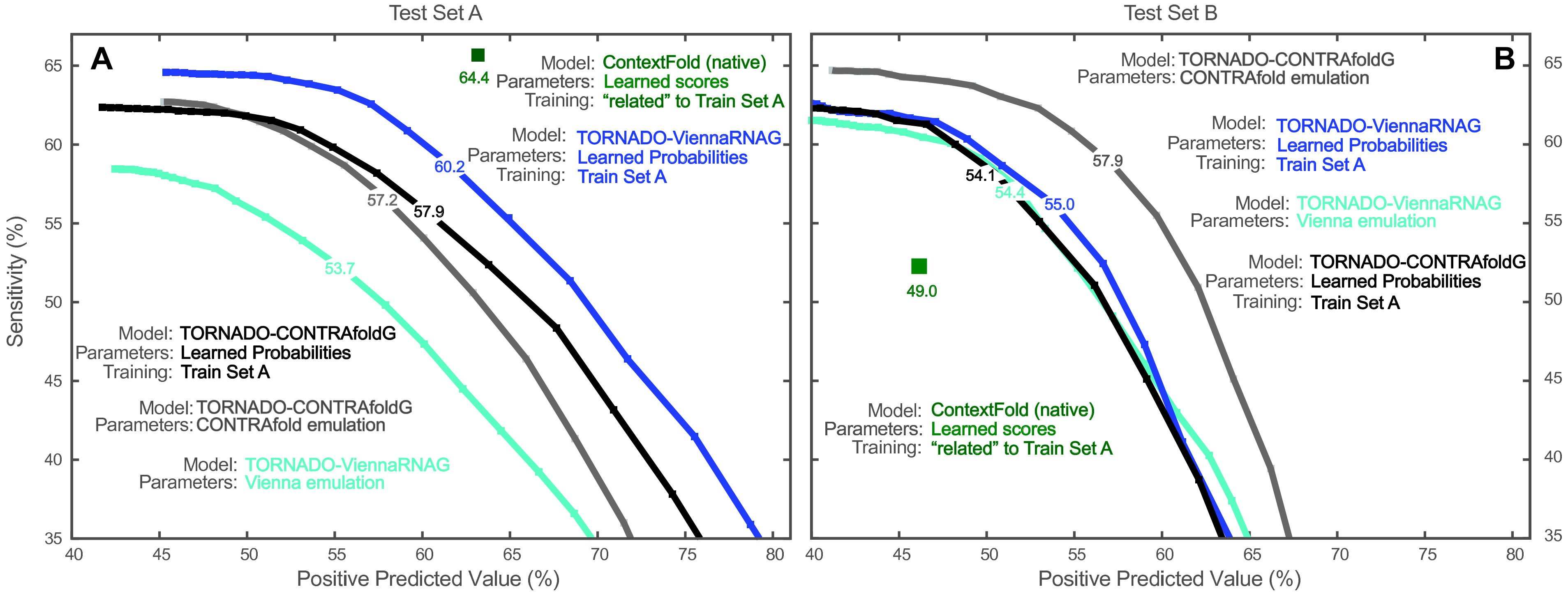
430 Sequences
44 % basepaired
8.3 % non-canonical

5.8S rRNA	(14)
U1	(18)
U2	(45)
7 Riboswitches	(233)
9 Cis regulatory RNAs	(116)
2 Ribozymes	(3)
bacteriophage pRNA	(1)

Benchmark

need for structurally diverse training sets

TrainSetA



Grammar	TrainSetA		TrainSetB		TrainSetA + TrainSetB		TrainSetA + 2 * TrainSetB	
	set best-F %		set best-F %		set best-F %		set best-F %	
	TestSetA	TestSetB	TestSetA	TestSetB	TestSetA	TestSetB	TestSetA	TestSetB
g6	47.8	46.2	48.5	49.3	48.7	47.0	49.1	47.5
basic_grammar	56.7	53.6	47.5	54.6	57.0	56.5	56.9	56.5
CONTRAfoldG	57.9	54.1	44.4	56.1	58.4	57.4	58.3	58.6
ViennaRNAG	60.2	54.4	42.8	56.0	60.4	57.7	60.2	59.4

A gradation of SCFGs

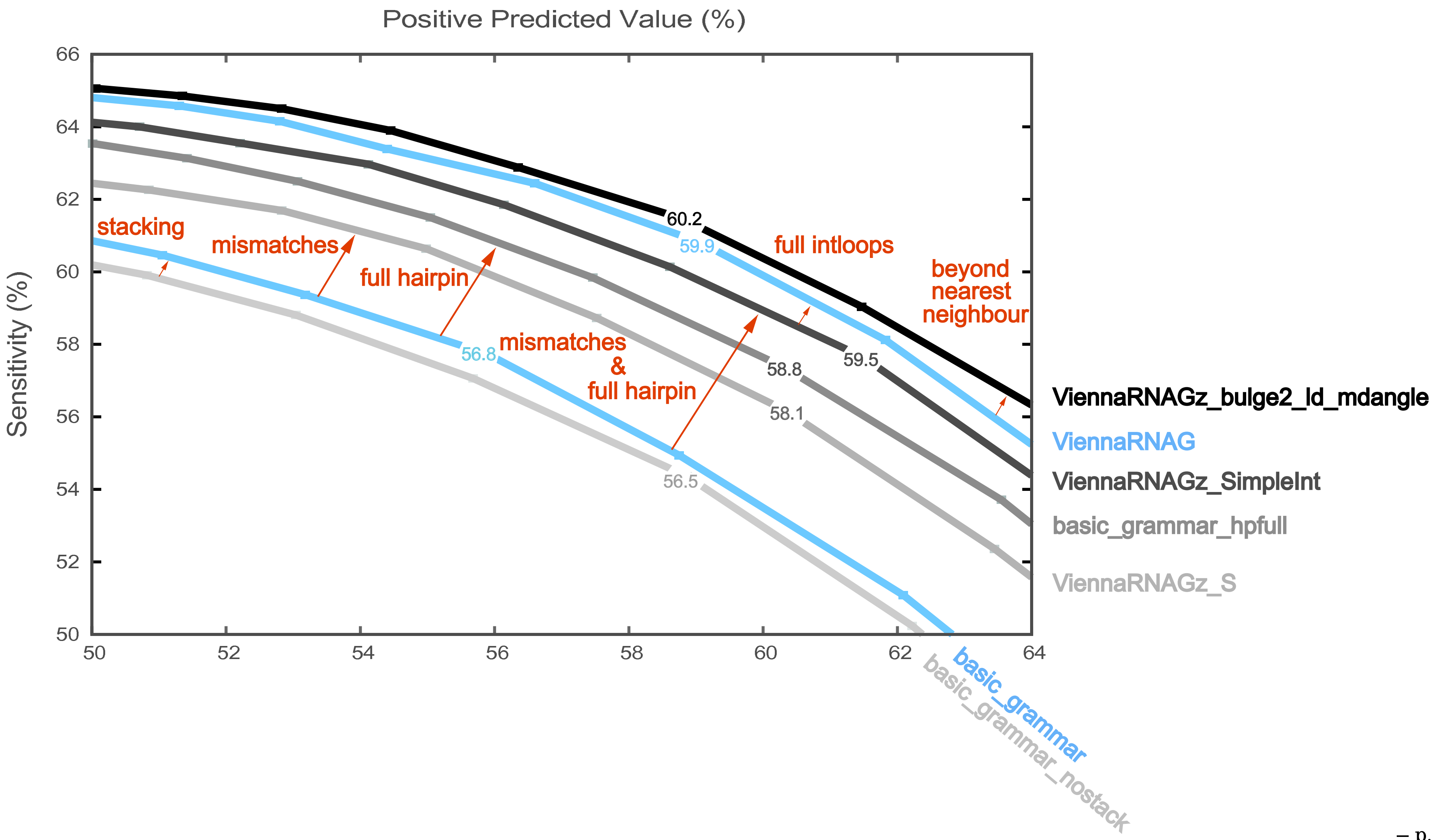
exploring different structural features

Grammar	Total Free Tied Parameters		Remarks
	4x4 bps	6 bps	
g6	21	11	Pfold grammar
g6s	261	41	Pfold + stacking
g6_stem	294	74	Pfold + stacking + helix length dist.
basic_grammar_nostack	572	532	loop length dist.
basic_grammar	1,022	582	loop length dist + stacking.
basic_grammar_dangle	1,143	643	basic_grammar + dangles
ViennaRNAGz_S	1,862	892	ViennaRNAGz_SimpleInt without tetraloops
CONTRAfoldGS	2,101	811	CONTRAfoldG with simpler 1nt bulges
basic_grammar_hpfull	5,342	2,202	basic_grammar + hairpin tetraloops + hairpin closing mismatches
CONTRAfoldG	5,448	1,278	CONTRAfold emulation
ViennaRNAGz_SimpleInt	6,105	2,495	ViennaRNAG minus 2x2,2x1 Internal loops
ViennaRNAGz_nostack	90,497	14,257	ViennaRNAG minus stacking
ViennaRNAG	90,947	14,307	ViennaRNA emulation
ViennaRNAGz_stem	90,980	14,340	ViennaRNAG + stem length dist.
ViennaRNAGz_bulge2	91,670	14,400	ViennaRNAG + explicit 1,2 bulges
ViennaRNAGz_ld	91,012	14,374	ViennaRNAG + all emissions by length dist
ViennaRNAGz_mangle	91,187	14,397	ViennaRNAG + multiloop mismatches
ViennaRNAGz_bulge2_ld_mdangle	91,977	14,557	ViennaRNAG + explicit 1,2 bulges + + all length dist + multiloop mismatches

Contribution of different features

Training: TrainSetA + 2*TrainSetB

Testing: TestSetA + TestSetB



Remarks

SCFGs have same expressive power than other statistical non-probabilistic model.

SCFGs have the advantage of easier training.

Training of complex models requires more structural diversity.

Lack of data:

Rfam: predicted structures, alignment structures

Protein data base: few and short sequences (compaRNA, 251 unique sequences, half of them shorter than 33 nts).

A dedicated effort to crystallize diverse structures

Beyond Watson-Crick pairs

in a motif-independent fashion

Would like to have alignments or single sequence annotation of non-WC basepairs.

Then, convert the unpaired “loop emissions” into a grammar of non cannonical pairs.

Assumptions:

One can extract paired preferences for a given pairing type independently of the RNA motif in which they happen.

Ignores stacking

This is “unprofiled” could allow for the identification of novel motifs