Using Tree Edit Distance as an RNA Secondary Structure Similarity Metric

Jon-Michael Deldin

Dept. of Computer Science
University of Montana
jon-michael.deldin@umontana.edu

2011-05-03

Outline

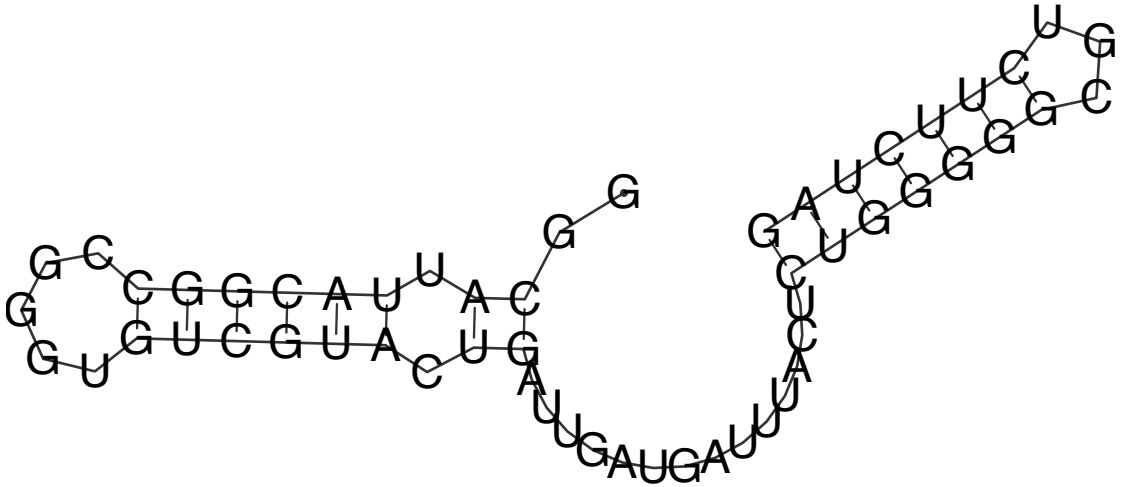
1. Project
2. Trees
3. Current Work
4. Future Work

Topic

1. Project
2. Trees
3. Current Work
4. Future Work

Goals

1. Represent an RNA secondary structure as a tree data structure
2. Determine the similarity between two structures using tree edit distance



Similarity

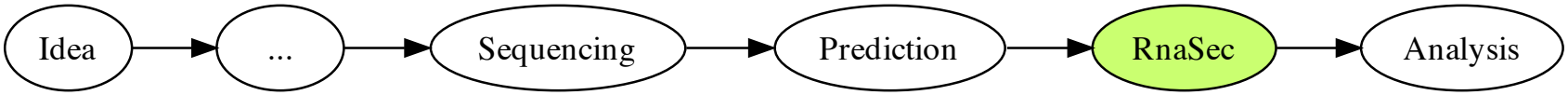
1 vs. 2

1 - /tmp		2 - /tmp
Chuck Norris won American Idol using only sign language.	1 →	Mr. T won American Idol using only sign language.
Chuck Norris has already been to Mars; that's why there are no signs of life.	2 →	Mr. T has already been to Mars; that's why there are no signs of life.
Chuck Norris can cut through a hot knife with butter	3 →	Mr. T can cut through a hot knife with butter
Chuck Norris is the reason why Waldo is hiding.	4 →	Mr. T is the reason why Waldo is hiding.
Chuck Norris can outrun his own shadow.	5 →	Mr. T can outrun his own shadow.
Chuck Norris counted to infinity - twice.	6 →	Mr. T counted to infinity - twice.

status: 6 differences

Actions

Pipeline



Data

- 100 SELEX aptamers (M. Ellenbecker, J.M. Lanchy, & J.S. Lodmell)
- Random RNA sequences

```
>MBE2A
GGCATTACGGCCGGG TGTCGTACTGATTGATGATTTACTCTGGGG GCGTCTTCTAG
```

Topic

1. Project
2. Trees
3. Current Work
4. Future Work

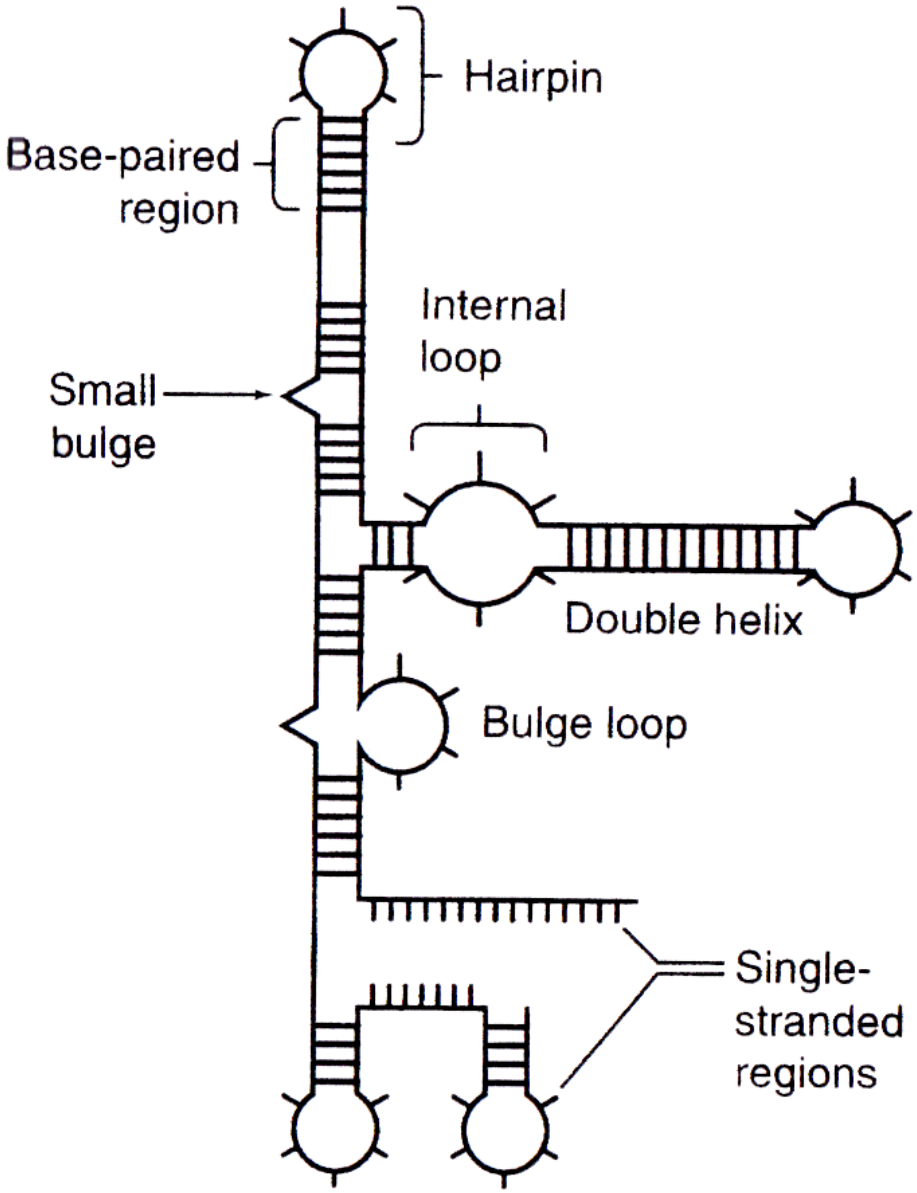
Trees

Definition (Tree Data Structure)

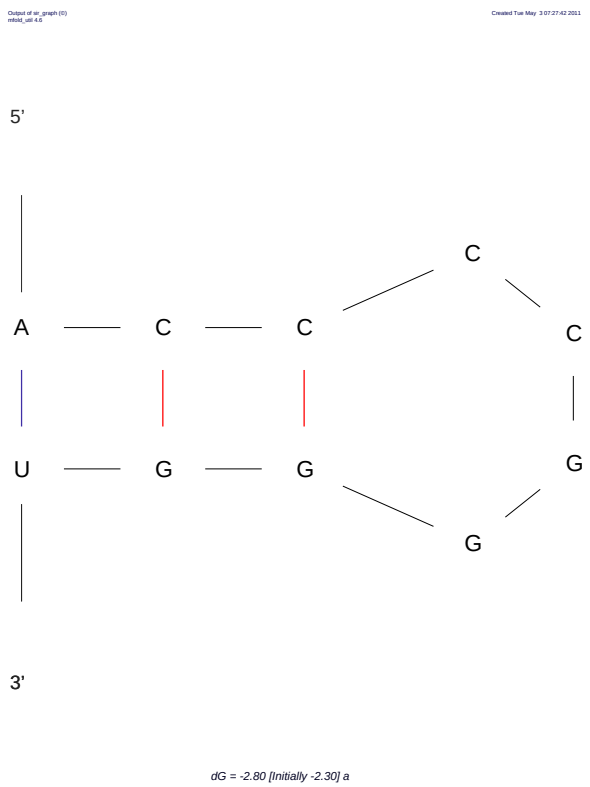
An abstract datatype that represents a hierarchy of objects (e.g., family tree). It is an ordered, directed, acyclic graph.

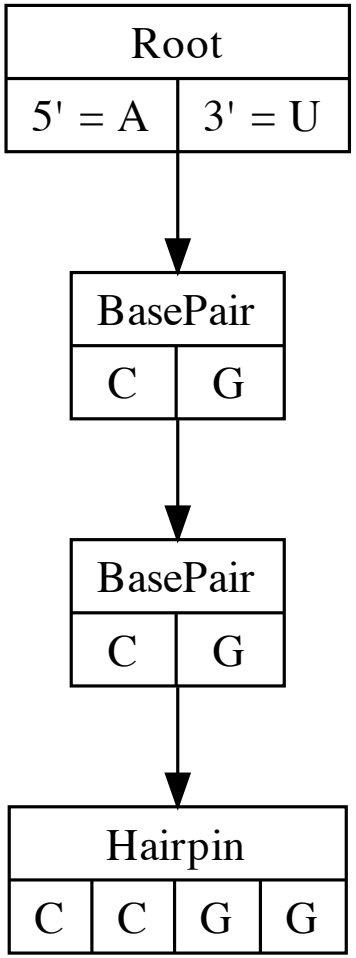
- Root node
- Traversal order matters (e.g., $5' \rightarrow 3'$)

Secondary Structure



Representation





Tree Operations

- Pruning
  - Removing nodes from a tree
  - Rules
    - If deleting a nt in a BP connected to a loop, move a nt from out of the loop
    - Cannot delete a single BP in a series of BPs (gap?)
- Grafting
  - Inserting nodes into a tree

Edit Distance

- Two different trees T_0 and T_1
- How many operations (insertions/deletions) are needed until $T_0 = T_1$?
- Not easy

Topic

1. Project
2. Trees
3. Current Work
4. Future Work

Genetic Algorithm

Implementing a genetic algorithm

- Take an RNA secondary structure
- Permute it (rearrange/insert/delete nodes)
- fitness(): BP score to original structure

Genetic Algorithm

Implementing a genetic algorithm

- Take an RNA secondary structure
- Permute it (rearrange/insert/delete nodes)
- fitness(): BP score to original structure

Chromosome	Secondary structure
Genes	Hairpins, bulges, base pairs

Genetic Algorithm

1. Permute ancestor n times or randomly generate a population of size n

Genetic Algorithm

1. Permute ancestor n times or randomly generate a population of size n
2. Until a solution is found:

Genetic Algorithm

1. Permute ancestor n times or randomly generate a population of size n
2. Until a solution is found:
    1. Get the fitness of all chromosomes

Genetic Algorithm

1. Permute ancestor n times or randomly generate a population of size n
2. Until a solution is found:
    1. Get the fitness of all chromosomes
    2. Select 2 parent chromosomes, x_0, x_1

Genetic Algorithm

1. Permute ancestor n times or randomly generate a population of size n
2. Until a solution is found:
    1. Get the fitness of all chromosomes
    2. Select 2 parent chromosomes, x_0, x_1
    3. offspring $\leftarrow$ crossover(x_0, x_1) according to a crossover rate

Genetic Algorithm

1. Permute ancestor n times or randomly generate a population of size n
2. Until a solution is found:
    1. Get the fitness of all chromosomes
    2. Select 2 parent chromosomes, x_0, x_1
    3. offspring $\leftarrow$ crossover(x_0, x_1) according to a crossover rate
    4. Mutate the offspring according to a mutation rate

Genetic Algorithm

1. Permute ancestor n times or randomly generate a population of size n
2. Until a solution is found:
    1. Get the fitness of all chromosomes
    2. Select 2 parent chromosomes, x_0, x_1
    3. offspring $\leftarrow$ crossover(x_0, x_1) according to a crossover rate
    4. Mutate the offspring according to a mutation rate
    5. Add offspring to the population

Genetic Algorithm

1. Permute ancestor n times or randomly generate a population of size n
2. Until a solution is found:
    1. Get the fitness of all chromosomes
    2. Select 2 parent chromosomes, x_0, x_1
    3. offspring $\leftarrow$ crossover(x_0, x_1) according to a crossover rate
    4. Mutate the offspring according to a mutation rate
    5. Add offspring to the population
3. Print solution

Topic

1. Project
2. Trees
3. Current Work
4. Future Work

Future Work

Tree edit distance