Синтаксический анализ для поиска в метагеномных сборках

Автор: Семён Григорьев

Лаборатория языковых инструментов JetBrains
Санкт-Петербургский государственный университет
Математико-механический факультет

11 мая 2016г.

Введение

- Анализ метагеномных сборок
- Поиск подпоследовательностей (РНК)
  - Идентификация организмов в метагеномной сборке

Обзор

- Xander — HMM
  - $(w_i, w_j) \in E(CAG) \leftrightarrow (v_i, v_j) \in E(HMM); (u_i, u_j) \in E(DG)$ v_j — insert или match
  - $(w_i, w_j) \in E(CAG) \leftrightarrow (v_i, v_j) \in E(HMM); u_i = u_j$ v_j — delete
- Infernal — Ковариационые модели, линейный вход
- REAGO — Infernal на прочитанных линейных участках с последующес сборкой

Вторичная структура РНК

- Несёт полезную информацию для идентификации подпоследовательностей
- Может быть задана с помощью граммтики

```
stem<s> :
    A stem<s> U | U stem<s> A
  | G stem<s> C | C stem<s> G
  | s

folded: stem<A*[3..7]>
```

Грамматики для описания вторичной структуры tRNA

- Уточнение шаблона
  - Возможности языков описания грамматики: метаправила, повторения
  - Выход за пределы KC-грамматик
- Далее на грамматику можно “навесить” вероятности и т.д.

Синтаксический анализ метагеномной сборки

- Регулярное множество (R) = конечный автомат (M) = граф конечного автомата или просто граф (H)

Синтаксический анализ метагеномной сборки

- Регулярное множество (R) = конечный автомат (M) = граф конечного автомата или просто граф (H)
- Грамматика G задаёт для цепочек свойство выводимости: $S \Rightarrow_G^* \omega$
- Задача: найти все цепочки $\omega \in R : S \Rightarrow_G^* \omega$
- Метагеномная сборка представима в виде графа, который можно рассматривать как KA, в котором все вершины стартовые

Синтаксический анализ регулярных множеств

- Состояние синтаксического анализатора обнозначно задаётся конечным набором данных — дескриптором
  - $s \to Ab \cdot c$ — “ситуация”, “слот”
  - Позиция во входе
  - ...
- Имея дескриптор можно продолжить разбор с описанного им места
- Дескрипторы переиспользуются: дальнейший разбор не зависит от истории дескриптора (**контекстно-свободная** грамматика)

Синтаксический анализ регулярных множеств: детали

- Процесс анализа конечен
- Внутренние структуры аналогичны структурам, используемым в обобщённом синтаксическом анализе (GLR, GLL)
  - Структурированный в виде графа стек (GSS)
  - Сжатое представление леса разбора (SPPF)
  - Управляющие таблицы
- Полиномиальная сложность (?)

Предлагаемый подход

1. Описание вторичной структуры искомых цепочек в виде грамматики
2. Выделение из метагеномной сборки цепочек на основе вторичной структуры
3. Фильтация цепочек сторонним инструментом (Infernal?)

Текущие результаты

- Реализован алгоритм синтаксического анализа регулярных множеств
  - Адотирован к обработке метагеномной сборки
  - Возвращает координаты начала и конца “подозрительныз подцепочек”
- Реализован алгоритм поиска с использованием булевых грамматик в линейном входе

Планы

- Понять, что делать с 16s
  - Есть ли “характерные подцепочки” или надо искать всю 16s целиком
  - Где взять описание вторичной структуры
- Оценить степень сужения пространства поиска: сколько “подозрительных участков” найдено в сбоке
- Возвращать не координаты начала и конца, а цепочки целиком
- Оценить время работы инструмента

Контакты

- Почта: rsdpisuy@gmail.com
- Исходный код YaccConstructor: https://github.com/YaccConstructor