Sprawozdanie z laboratorium: Bioinformatyka (szablon)

Część I: Analiza teoretyczna

27 marca 2017

Prowadzący: prof. dr hab. inż. Marta Kasprzak

Autorzy:	**Damian Jurga**	inf.....	I2	jasiu@serwer.domena.poczta.pl
	Grzegorz Miebs	inf122453	I2	grzegorz.miebs@student.put.poznan.pl

Zajęcia środowe, 11:45.

Oświadczamy, że niniejsze sprawozdanie zostało przygotowane wyłącznie przez powyższych autorów, a wszystkie elementy pochodzące z innych źródeł zostały odpowiednio zaznaczone i są cytowane w bibliografii.

1 Wstęp

Celem tego sprawozdania jest przedstawienie teoretycznego opracowania metody heurystycznej rozwiązującej problem sekwencjonowania łańcuchów DNA z błędami pozytywnymi oraz negatywnymi w czasie wielomianowym. Alogrytm mając dany na wejściu zbiór S słów o długości l nad alfabetem $\{A, C, G, T\}$, długość n sekwencji oryginalnej, powinien zwrócić sekwencję o długości nie większej niż n zawierającą maksymalną liczbę słów z S.

2 Algorytm

2.1 Opis

Do rozwiązania tego problemu zbudujemy graf, którego wierzchołkami będą słowa ze zbioru S, a wartości łuku między wierzchołkami będą równe przesunięciu między tymi słowami. Przykładowo łuk z wierzchołka ACCGT do wierzchołka CCGTC będzie miał wartość 1 a do wierzchołka GTCGT wartość 3. Na skonstruowanym w ten sposób grafie rozwiązujemy problem komiwojażera maksymalizujący liczbę odwiedzonych wierzchołków przy ograniczeniu na sumę wartości wykorzystnych łuków, która nie może być większa od $n - l$, gdyż w przeciwnym razie na wyjściu otrzymamy sekwencję dłuższą niż oryginalna. Aby ograniczyć ponowne odwiedzanie tych samych wierzchołków, będziemy zwiększać wartość łuków prowadzących do odwiedzonych już wierzchołków o pewną stałą C. Do rozwiązania problemu komiwojażera posłużymy się przeszukiwaniem wiązkowym oraz algorytmem wspinaczki.

2.2 Lista kroków

1. Zbudowanie grafu
2. Zaczynamy z losowego wierzchołka
3. Znajdujemy k najbliższych miast
4. Do każdej z k dotyczasowych ścieżek liczymy odległość po dodaniu każdego z wierzchołków z uwzględnieniem kary za powtórne odwiedzenie tego samego wierzhołka.Wybieramy k najkrótszych ścieżek
5. Powtarzamy krok 4 aż koszt ścieżek przekroczy krytyczną wartość $n - l$
6. Dla każdej z k wygenerowanych ścieżek sprawdzamy które przestawienie parami pozwoli maksymalnie zmniejszyć koszt ścieżki i wykonujemy je
7. Powarzamy krok 6 k razy, bądź aż do uzyskania lokalnego optimum
8. Jeśli długość ścieżki jest mniejsza od $n-l$ o więcej niż z to ponownie wykonujemy przeszukiwanie wiązkowe w celu rozszerzenia aktualnych ścieżek. W przeciwnym wypadku dokonujemy pełnego przeglądu.

2.3 Złożoność obliczeniowa

Złożoność pierwszego przeszukiwania wiązkowego jest równa $O(k * |S|^2)$
Złożoność algorytmu wspinaczkowego $O(k * |S|^2)$
Złożoność drugiego przeszukiwania wiązkowego $O(k * z * |S|)$
Złożoność całego algorytmu $O(k * |S|^2)$

3 Usprawnienia

3.1 brak błędów negatywnych

Gdyby założyć, że w danym zestawie danych jedyne błędy jakie występują to błędy pozytywne, moglibyśmy wówczas podczas przeszukiwania wiązkowego wycofywać się z wierzchołków, które są odległe o więcej niż 1 od każdego innego wierzchołka. Pozwoliłoby to ominąć część błędów pozytywnych. Moglibyśmy także brać pod uwagę tylko i wyłącznie łuki o wartości 1, jednak wówczas algorytm przeszukiwania wiązkowego mógłby wielokrotnie przechodzić po tych samych łukach, odwiedzając kilkukrotnie te same wierzchołki, pomijając jednocześnie inne. Z tego powodu sensowne wydaje się pozostawienie także łuków o większych wartościach. Próg powyżej którego łuki nie będą brane pod uwagę, zwiększający skuteczność algorytmu będzie zależny od topologii sieci utworzonej z łuków o wartości mniejszej bądź równej temu progowi. Jeśli sieć ta będzie odpowiednio gęsta, wówczas szansa na wielokrotne odwiedzanie tych samych wierzchołków spad. Optymalną gęstość takiej sieci najłatwiej będzie ustalić eksperymentalnie.

3.2 brak błędów pozytywnych

Przy założeniu, że w danym zestawie danych występują tylko i wyłącznie błędy negatywne algorytm powinien odwiedzić wszystkie wierzchołki, dlatego stała C, karająca za ponowne odwiedzenie tego samego wierzchołka może być wówczas znacznie wyższa, co spowoduje, że algorytm przeszukiwania wiązkowego znacznie rzadziej, bądź nigdy będzie odwiedzał te same wierzchołki, jeśli będą dostępne wierzchołki do tej pory nieodwiedzone. Nawet jeśli spowoduje to, że ścieżka przebiegać będzie w sposób odległy od optymalnego, istnieje spore prawdopodobieństwo, że zostanie ona poprawiona w drugim etapie czyli algorytmie wspinaczki.