Zadanie 1 z listy 7 - “Kompresja Danych”

Łukasz Klasiński

16 maja 2020

Zadanie 1

Omów różnice między wariantami *ppma*, *ppmb*, *ppmc* algorytmu *ppm*. Przedstaw sposób implementacji tych algorytmów w czasie liniowym ze względu na długość tekstu przy założeniu, że rozmiar alfabetu i maksymalny rozmiar kontekstu są stałe.

Rozwiązanie

Te warianty głównie różnią się tym, z jakim prawdopodobieństwem mają występować znaki `Escape`.

PPMA

$$p(Escape) = \frac{1}{m+1}$$
$$p(a_i) = \frac{c_i}{m+1}$$

oznaczenia:

p - szacowane prawdopodobieństwo

m - liczba symboli, które wystąpiły w danym kontekście

c_i - liczba wystąpień symboli a_i w danym kontekście

Oznacza to, że w kontekście dowolnego rzędu większego niż -1, zawsze znajdzie się ten symbol `Escape`.

Implementacja

Kodujemy prawdopodobieństwa zgodnie z kontekstami identycznie jak w zwykłym *ppm*. Konteksty trzymamy w tablicach hashujących do których dostajemy się przez odpowiedni prefix. Następnie aktualizujemy prawdopodobieństwa (odwiedzając tą samą liczbę co przy ustalaniu prawdopodobieństw kontekstów). Jako że odwiedzimy co najwyżej wszystkie konteksty których jest ustalona stała ilość k, to taki algorytm wykona się w czasie $O(|w| * k) = O(|w|)$.

Wady tej wariancji - okazuje się że takie szacowanie symbolu wyjścia jest często nadmiarowe.

PPMB

$$p(Escape) = \frac{t}{m}$$
$$p(a_i) = \frac{c_i - 1}{m}$$

Oznaczenia:

p - oszacowane prawdopodobieństwo

m - liczba symboli, które wystąpiły w danym kontekście

t - liczba różnych symboli, które wystąpiły w danym kontekście

c_i - liczba wystąpień symboli a_i w danym kontekście

Widzimy zatem, że symbole, które wystąpiły w kontekście tylko raz są traktowane tak jakby wcale w nim nie występowały - dopiero je śli symbol wystąpił w kontekście 2 razy, to jest on w nim kodowany. Zatem jeśli alfabet jest duży, to dwukrotne kodowanie symbolu jako nowego w danym kontekście może znacząco pogorszyć stopień kompresji. Widać również, że prawdopodobieństwo wystąpienia symbolu `Escape` jest mniejsze niż w przypadku $PPMA$.

Implementacja

Tak samo jak $PPMA$, tylko inaczej dobieramy prawdopodobieństwa w kontekstach.

PPMC

$$p(Escape) = \frac{t}{m}$$

$$p(a_i) = \frac{m-t}{m} \cdot \frac{c_i}{m}$$

Oznaczenia, jak w $PPMB$.

Dodatkowo mamy dwa etapu kodowania - * Pierwszy etap - kodujemy binarną flagę, czy jest symbol wyjścia czy nie * Drugi etap - kodowanie symbolu jeśli nie było symbolu wyjścia, bądź zejście do niższego modelu jeśli był symbol wyjścia.

Osiągamy znacznie lepsze współczynniki kompresji niż w metodach $PPMA$ i $PPMB$. Problemem jest fakt, że podczas liczenia prawdopodobieństw mamy w mianowniku m^2, co przy dłuższych tekstach może spowalniać liczenie floatów.

Implementacja

Algorytm poza tym, że ma etapy niczym nie różni się od $PPMB$.