Algoritmus Re–Pair

Moravec Vojtěch

LS 2020

1 Popis algoritmu Re–Pair

Algoritmus Re–Pair, poprvé představen v článku [1], je kompresní algoritmus založený na bezkontextové gramatice. Na vstupu dostává tento algoritmus řetězec znaků, např. text, který převede na řetězec terminálních a neterminálních symbolů bezkontextové gramatiky. Autoři zařadili tento algoritmus do skupiny *off-line* slovníkových kompresních metod.

Off-line metody využívají celého vstupního řetězce k vytvoření slovníku. Tyto metody tedy musí načíst celý vstupní soubor nebo jeho část do paměti. Části se stále rozumí mnohem větší část, než využívají *on-line* metody. Algoritmy v *on-line* skupině, tvoří slovník pomocí té části vstupu, kterou již přečetli, jak je tomu například u slovníkových kompresí LZ. Nevýhodou *off-line* metod je tedy velká paměťová náročnost. Oproti tomu výhodou, důvodem, proč vůbec uvažovat o této skupině, je možnost vytvořit slovníkové fráze, vedoucí k lepším kompresním výsledkům. Díky znalosti vstupních dat jako celku známe obecně lepší souvislosti.

V tomto dokumentu budeme popisovat kompresi textu a očekáváme od čtenáře základní znalost bezkontextové gramatiky. Algoritmy využívají bezkontextové gramatiky převádějí vstupní text na sekvenci terminálních a neterminálních symbolů. Aby mohli provést tuto transformaci musí nalézt přepisovací pravidla, které budou vést k maximální kompresi. Ukázku, jak můžeme transformovat text `abrakadabra` pomocí bezkontextové gramatiky nalezneme v Tabulce 1. Záměrně jsme využili přepisovací pravidla, které mají na pravé straně pouze dva symboly.

Text	Pravidla
`abrakadabra`	
`ArakadAra`	$A \to ab$
`BakadBa`	$B \to Ar$
`CkadC`	$C \to Ba$

Tabulka 1: Ukázka přepsání textu podle bezkontextové gramatiky

Je tedy zřejmé, že abeceda vzniklá na konci komprese, bude oproti terminálním znakům z textu, obsahovat také nové neterminální symboly. Výsledkem algoritmu je tedy řetězec a množina přepisovacích pravidel. Obě tyto informace musí být uloženy v komprimovaném souboru, nebo přeneseny přes síť. K zakódování se většinou využívá různých entropických metod, generující kódy proměnlivé délky, jako je například Huffmanovo kódování nebo Aritmetické kódování.

Samotná název Re–Pair je zkratkou anglického *Recursive Pairing*, který už sám o sobě napovídá, že se bude jednat o metodu využívající určité rekurze a párování. Originální algoritmus vždy nahrazuje nejčetnější pár v textu. Párem rozumíme dvojici terminálních nebo neterminálních symbolů. Označíme-li pomocí malých písmen terminální symboly a pomocí velkých písmen neterminální symboly, tak mohou nastat čtyři situace:

- ab
- AB
- aB
- Bc

Algoritmus nijak neupřednostňuje žádnou z nich. Nahrazování probíhá do té doby, dokud se v nahrazeném textu nachází určitý pár alespoň dvakrát. Postupně upravovaný text je dobrý způsob, jak si představit fungování algoritmu, ale při samotném běhu není žádné nahrazování prováděno. Dále bude popsáno, jak dokáže algoritmus postupně redukovat text a přitom vědět, v jakém stavu se text zrovna nachází.

Důvodem
huuuaaa

Reference

[1] N. J. Larsson and A. Moffat, "Off-line dictionary-based compression," *Proceedings of the IEEE*, vol. 88, no. 11, pp. 1722–1732, 2000.