研　究　報　告　書

「圧縮データ索引に基づく巨大文書集合からの関連性マイニング」

研究タイプ：通常型
研究期間：　平成 21 年 10 月～平成 25 年 3 月
研 究 者：　坂本　比呂志

1．　研究のねらい

データ圧縮の研究は，1948 年のクロード・シャノンの論文から始まったといわれている．この論文によって情報理論が創始され，文字の生成確率によって定まるエントロピーと呼ばれる概念によって，データ圧縮を定量的に評価できるようになった．その後，ユニバーサル符号などのアルゴリズムの登場で，データ圧縮の研究が盛んになった．本研究では，大量データ時代の新しいデータ圧縮を実現するための基盤技術の創出を目指す．通常データ圧縮といえばデータサイズを小さくし，ネットワークのスループットや記憶領域の消費量を改善することが目的であると考えられているが，1990 年代の初め頃にはじまった先駆的な研究により，データ圧縮の分野に新しい潮流が生まれている．すなわち，データを圧縮することで様々な計算を高速化する「圧縮情報処理」である．高速化可能な計算としては，圧縮された状態の文書群に対する全文検索や部分文書の高速復元などがあり，これらの技術がさまざまな分野で応用されている．しかし，これらの圧縮情報処理は，静的で比較的小規模のデータに対しては適用可能であるが，ストリームデータのように大量で変化し続けるデータへの適用は困難であった．また現代は，圧縮が必要とされている大規模データ群，高速処理のためのデータ構造とアルゴリズム，圧縮情報処理を可能にする高性能なハードウェアのすべてが揃っている時代でもある．そこで本研究では，圧縮情報処理を大規模データストリームに適用可能な新しい技術を実現し，大規模データゆえに発展が阻害されている分野へこの技術を輸出することで，圧縮情報処理の社会応用を目指す．特に，大規模な計算機環境を利用することが困難な分野や研究課題に対する貢献を目標とし，「データは圧縮して利用するものである」という理念を世界に浸透させる．

2．　研究成果

（1）概要

これまでの圧縮情報処理は，BWT　や接尾辞配列と呼ばれる文字列のソートに基づく手法が主流で，これらの手法では，同じ接尾辞を持つ部分文字列同士が近くになるように原データを並べ替えるので，圧縮率はよい反面，前処理のためのコストが大きく，また，データが動的に変化するような場合には適用できない．一方，文字列圧縮の世界では，2000 年に提案された文法圧縮と呼ばれる手法が注目を集めている．これは，データを一意に導出する文脈自由文法(CFG)の構文木を構築し，共通の部分木を削除することで圧縮を達成する手法である．文法圧縮はその名が示すとおり，自然言語の構文木を模倣したもので，本来文法構造を持たない DNA シークエンスや文字の羅列に対しても擬似的な文法を構築することで，任意の文字列の圧縮を可能とする．与えられた文字列に対して最小の CFG を構築する問題は，NP−困難と

いう現実的な時間では計算できないと考えられている問題のクラスに含まれるが，ほぼ最適解を線形時間で計算できるアルゴリズムが 2002～2003 年にかけて本さきがけ研究者を含む複数の研究者によって発見された．文法圧縮は，それまでの主要な圧縮法と比較して圧縮率は劣るものの，データの追加に頑健であり，省スペースな領域でも動作するという特徴を持っている．したがって，それまではメモリ上で直接扱うことができなかった規模のデータを圧縮することが可能となった．本研究では，この文法圧縮をさらに発展させ，全文検索や部分文書展開の機構を組み込んだコンパクトな索引構造を構築し，実証実験によってその性能を確認した[原著論文 2 ,5]．この索引構造はそれまでにないタイプのものであり，特に，Web のアーカイブのように冗長なデータをノート PC のような非常に制限された環境下で極限まで圧縮することができる．本研究では，このデータ構造とアルゴリズムをさらに拡張し，ストリーム型データに適用可能とした[原著論文 3]．このアルゴリズムにより，メモリ効率を保ったまま，データをオンライン圧縮できるため，ソーシャルネットワークを流れる巨大なデータに対してもダイナミックな検索が可能となった[原著論文 1,4]．このアルゴリズムの応用として，巨大な twitter データに対して，ノート PC 上で索引構築から頻出パターン発見までが可能となることを示した．以上のように，本研究では，文法圧縮の可能性を広げ，大規模データストリームマイニングに対する基盤技術を開発した．

（2）詳細

研究テーマA 「データ圧縮の理論に基づくコンパクトな索引構造の構築」

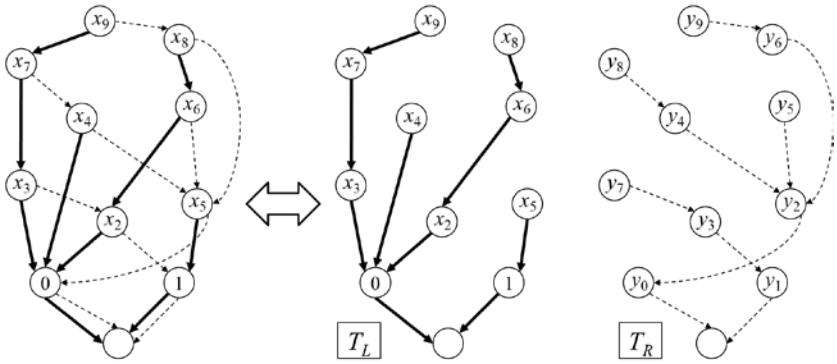
文法圧縮の理論を発展させ，省スペースでストリーム処理可能なオンラインアルゴリズムを構築した．提案手法は，これまでの符号化方と比較して，符号化に要する領域計算量を半分程度に抑えつつ，出力サイズを十分に小さくすることを可能とした．以下の表は，従来手法との出力サイズ(dictionary size)，メモリ使用量(overhead)，計算時間(sec)の比較である．

wikipedia (5.5GB) method	dictionary size (MB)	overhead (MB)	time (sec)
proposed	3,367	5,748	4,903
dag_vector	10,014	12,511	16,960
STL vector	9,401	11,898	3,125
genome (3.2GB) method	dictionary size (MB)	overhead (MB)	time (sec)
proposed	1,819	3,114	2,417
dag_vector	5,104	6,461	6,359
STL vector	5,109	6,467	1,576

研究テーマB 「グラフ構造の高速参照と並列化による大規模化」

圧縮データ上で全文検索を実現するためには，補助データ構造を用いた二分探索を実行する必要がある．この補助データ構造が大きすぎる場合，原データサイズを超えてしまい，圧縮索引の意味がなくなってしまう．本研究では，この補助データ構造を含めても文法圧縮のサイ

ズとほぼ変わらないデータ構造を実現した．それが図の文法圧縮の全域木分解である．一般に，文法圧縮はある種の DAG と呼ばれるグラフ構造と等価であり，それは 2 つの全域木に分解できる．木構造は簡潔データ構造と呼ばれる情報理論的下限まで小さく符号化することが可能なため，索引構造全体をコンパクトに表現できる．この符号化手法は本研究で初めて提案された．

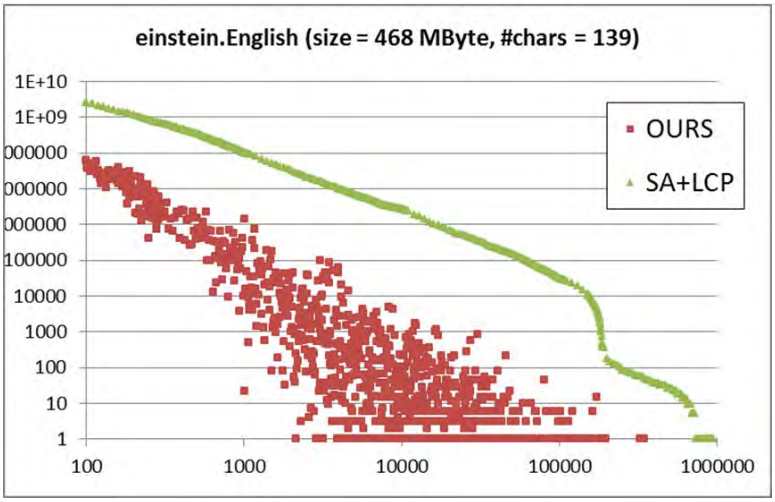


文法圧縮の全域木分解

研究テーマC 「圧縮データ索引による重要情報のマイニング」

文法圧縮によって複数のパターンが 1 つの変数(すなわち CFG の非終端記号)として符号化されているとき，その変数を取得して，元のパターンに復元することで，頻出なパターンを近似的に発見しているといえる．本研究では，この文法圧縮上でのパターン発見問題を定式化し，頻出パターンの近似発見問題を理論的に解析した．そしてこのアルゴリズムを大規模実データへ適用することで，実際のパターン発見に適用した．下の表は，Twitter データからの繰り返しパターンの抽出結果である．データサイズは 30GB であり，通常の検索手法ではこの数倍のメモリが必要となるが，本手法では，圧縮後のサイズ 14GB 程度のメモリがあれば実行可能となり，ノート PC 上でこの問題を取り扱うことが可能である．また，グラフの散布図は，パターン検出の近似がどの程度成功しているかを表している．SA+LCP は接尾辞木と高さ配列を用いた厳密解の個数であるが，データサイズの数十倍の領域を必要とする．OURS は本手法で取得したパターンであり，厳密解を約一桁下回るが，線形時間かつ圧縮サイズと同領域で実行可能である．

Text size	Index 構築時間	頻出パターン発見時間	Index size	出力数（閾値：80Byte）
約 30 GB	42,956 sec (約12時間)	2,034 sec (約30分)	約 14 GB	55,750,313



3. 今後の展開

今後は，本研究で提案した文法圧縮とその応用の利点である，省スペースとオンライン性を最大限に生かした社会応用を目指す．具体的には，SNS に代表されるストリームデータからリアルタイムに情報を取得するための基盤技術として売り出していきたい．また，この分野を発展させるために，「圧縮情報学」のセミナーを開催し，若手の育成に努める．

4. 自己評価

本研究の成果は，文法圧縮の理論をほぼ完成させたこと，プログラムを公開して他の研究者のツールを提供したこと，応用例を通して今後の可能性を示せたことである．一方，当初の目標であった，文書群からのマイニングと並列化による大規模化には手が届かなかった．これらは自然言語処理や Hadoop などの専門技術が必要なため，引き続き共同研究として追求していく．

5. 研究総括の見解

文字列共起を利用する超大規模なテキストなどの生文書の類似性計算、文書分類、検索に関する新しい技術の開発がテーマである。

さきがけ研究を通じ、文法圧縮の理論をほぼ完成させ、プログラムを公開して他の研究者に提供し、応用例を通して今後の可能性を示しており、評価できる。文書群からのマイニングと並列化による大規模化は果たしていないが、データを圧縮したままでの情報処理の道を切り開いており、評価できる。

「圧縮情報学」を提唱し、この分野を発展させたいとしており、今後に期待している。

6. 主な研究成果リスト

（1）論文（原著論文）発表

1. Scalable Detection of Frequent Substrings by Grammar-Based Compression, M.Nakahara, S.Maruyama, T.Kuboyama, H.Sakamoto,

IEICE Trans. on Information and Systems, in press
2. ESP-Index: A Compressed Index Based on Edit-Sensitive Parsing, S.Maruyama, M.Nakahara, N.Kishiue, H.Sakamoto, Journal of Discrete Algorithms 18:100-112 (2013)
3. An Online Algorithm for Lightweight Grammar-Based Compression, S. Maruyama, H. Sakamoto, M. Takeda, Algorithms 5(2):214-235 (2012)
4. Extracting research communities from bibliographic data, Y.Nakamura, T.Horiike, T.Kuboyama, H.Sakamoto, KES Journal 16(1): 25-34 (2012)
5. Context-sensitive grammar transform: compression and pattern matching, S.Maruyama, Y.Tanaka, H.Sakamoto, M.Takeda, IEICE Trans. on Information and Systems, E93-D(2):219-226(2010)

（2）特許出願

研究期間累積件数：0 件

（3）その他の成果（主要な学会発表、受賞、著作物、プレスリリース等）

1. Pattern Extraction from Graphs and Beyond,
H. Sakamoto, T. Kuboyama,
Book chapter in Multimedia Services in Intelligent Environments,
Springer, to appear

2. Variable-Length Codes for Space-Efficient Grammar-Based Compression,
Y. Takabatake, Y. Tabei, H. Sakamoto,
19th International Symposium on String Processing and Information Retrieval,
398-410(2012)

3. Scalable Detection of Frequent Substrings by Grammar-Based Compression,
M.Nakahara, S.Maruyama, T.Kuboyama, H.Sakamoto,
The 14th International Conference on Discovery Science, 236-246(2011)

4. ESP-Index: A Compressed Index Based on Edit-Sensitive Parsing,
S.Maruyama, M.Nakahara, N.Kishiue, H.Sakamoto,
18th International Symposium on String Processing and Information Retrieval,
398-409(2011)

5. 新聞報道：平成21年12月8日付　毎日新聞「理系白書'09」