クエリ中の語を含むことを保証するクエリフォーカス要約

西野 正彬† 安田 宜仁† 平尾 努‡ 鈴木 潤‡
† NTT サイバーソリューション研究所 ‡ NTT コミュニケーション科学基礎研究所
{nishino.masaaki,yasuda.n,hirao.tsutomu,suzuki.jun}@lab.ntt.co.jp

概要

クエリフォーカス要約とは，文書自動要約の一種であり，ドキュメント集合とともに与えられたクエリを加味して要約を生成することを特徴とする．既存のクエリフォーカス要約生成手法では，クエリ中に含まれる語が生成される要約中に含まれる可能性は高いものの，必ずしも要約に語が含まれることを保証することはできなかった．本稿では，生成された要約文書がクエリ中の語を含むことを制約とした最適化問題としてクエリフォーカス要約を定式化する．そのうえで，ラグランジュ緩和を用いて解を求めることで，制約を満たす解を効率的に求めることを可能とする．

1 はじめに

クエリフォーカス要約 *(query-focused summarization)* とは，要約の対象となる文書とともに与えられたクエリを反映した要約を作成することである．クエリフォーカス要約は，例えば情報検索において，与えられた検索クエリに対する検索結果を提示する際に，提示する文書の概要を示す目的で利用される．

これまで，自動要約生成問題は，生成される要約の文字数に制限があるもとでの最適化問題として定式化され，解かれることが多かった．クエリフォーカス要約についても同様であり，クエリに基づいて文のスコアを定めた後に，スコアを最大化する文集合を生成する問題として定式化できる．

その一方で，クエリフォーカス要約には，一般の文書要約と異なり，生成された要約中でのクエリ中に含まれる語の役割が重要になるという違いがある．例えばWeb検索におけるスニペットを表示するためにクエリフォーカス要約を提示する際には，生成された要約にクエリ語が含まれることが望ましいだろう．しかし，これまでのクエリフォーカス要約手法では，クエリ中の語が要約に含まれる可能性は高いものの，よりスコアの高い語があればそちらが選ばれるため，必ずクエリ中の語が出現することを保証することができなかった．一方でクエリ中の語が出現している文のみを用いて要約を構成すると，今度は要約対象の文書群のトピックに関する内容が含まれなくなるという問題があった．

本稿では，クエリ中に含まれる語が，生成される要約に含まれることを保証するクエリフォーカス要約生成手法を提案する．手法の特徴は，通常の要約問題を最適化問題として解く際に，要約の文字数の制約のほかに，クエリ中の語が生成される要約に必ず含まれるという制約を加えて問題を解く点にある．ラグランジュ緩和を用いて，追加した制約のもとでの最適化を行うことによって，制約を用いなかった場合と同程度の計算時間で求める結果を得ることが可能となる．

2 関連研究

クエリフォーカス要約については，Tombrosらによる手法 [Tombros 98] 以降，様々な手法が提案されている．Daumé による BAYESUM[Daumé 06]，および [Tang 09] は，LDA(Latent Dirichlet Allocation) を拡張したモデルを用いた要約方法を提案している．既存のクエリフォーカス要約手法は，クエリと似た文のスコアを高く設定することによって，クエリに関連する文が要約に出現しやすいようにしている．しかし，あくまでスコアを高く設定するのみであり，生成される要約中にクエリ中の特定の語が含まれることは保証されない．

提案手法で利用するラグランジュ緩和は，組合せ最適化問題の古典的な解法として知られている [Korte 08]．ラグランジュ緩和およびその特殊なケースである双対分解が，近年自然言語処理のさまざまなタスクに適用され成果を挙げている [Rush 11][Chang 11][Rush 10][Koo 10] .

3 クエリフォーカス要約

クエリフォーカス要約を定式化する．入力として，文書の集合 D とクエリ q が与えられる．D は適切な前処理によって構成単位の集合 $D = \{s_1, \ldots, s_N\}$ に分割される．構成単位としては，例えば語や文，パラグラフなどがある．要約とは，D から適切な構成単位を選択して，部分集合 $S \subseteq D$ を構成する問題と捉えることができる．以下では各構成単位がそれぞれ一つの文であるとして説明を続ける．クエリ q としては複数の単語や質問文等が利用され得る．

なお，以下では任意の文の集合 $S \subseteq D$ を N 次元の二値ベクトル $\mathbf{y}$ を用いて表す．すなわち，ある文 s_i について，$s_i \in S$ ならば $y_i = 1$，そうでないなら $y_i = 0$ であるベクトルとして表現する．クエリフォーカス要約は，長さの条件のもとで最大のスコアを与える $\mathbf{y}^*$ を求める問題として，以下のように定式化できる．

$$\mathbf{y}^* = \arg\max_{\mathbf{y}} f(\mathbf{y}, q) = \sum_{i=1}^{N} w_i(q) y_i \tag{1}$$

$$\text{subject to } \sum_{i=1}^{N} l_i y_i \leq L_{\max} \tag{2}$$

ここで $L_{\max}$ は要約の文字数の上限である．l_i は文 s_i の文字数を表す．q は入力として与えられたクエリである．$w_i(q)$ は文 s_i スコアであり，文 s_i に含まれる単語の重要度の和と，文とクエリとの類似度の和として

$$w_i(q) = \sum_{t \in s_i} \mathrm{tfidf}(t) + \mathrm{sim}(q, s_i)$$

として定義する．$\mathrm{tfidf}(t)$ は単語 t の tf·idf 値，$\mathrm{sim}(s_i, q)$ はクエリ q と文 s_i の tf·idf 値のコサイン類似度である．上記の定式化では，整数重みのナップサック問題として，効率的に厳密解を求めることが可能である．

4 語数の制約を満たす要約

前章での定式化のもとで要約を生成すると，クエリの影響が強い文ほど要約に含まれやすくすることが可能である．しかし，クエリ中の語が要約中に必ず含まれることを保証するものではない．そこで，本稿ではクエリ中の語を含むことを制約とする最適化問題として，自動要約生成を行う手法を提案する．前章で示したクエリフォーカス要約生成問題は，ナップサック問題として厳密解を求めることができた．しかし，クエリ中の語を含むことを制約として加えると，一般の組合せ最適化問題となるため，効率的に解を求めることができないという問題がある．そこで，本稿ではラグランジュ緩和を用いることで，ナップサック問題を繰り返し解くことで効率的に解を求める手法を示す．

制約の種類はいくつか考えられるが，ここではまず簡単な例としてクエリ中に含まれる単語のうち，いずれかが要約中に出現するという制約のもとで説明する．他の制約については後述する．クエリ中の語の出現を制約とした要約生成問題は以下のように定式化できる．

$$\begin{aligned} \mathbf{y}^* \quad &= \arg\max_{\mathbf{y}} f(\mathbf{y}, q) \\ \text{subject to} \quad &\sum_{i=1}^{N} l_i y_i \leq L_{\max} \\ &c_q(\mathbf{y}) \geq 1 \end{aligned}$$

ここで $c_q(\mathbf{y})$ は，クエリ中の単語を含む文の数を出力する関数であり，

$$c_q(\mathbf{y}) = \sum_{i=1}^{N} h_i y_i$$

と定義する．ここで $h_i \in \{0, 1\}$ であり，$t \in q$ かつ $t \in s_i$ である単語 t が存在するときに 1，そうでないときに 0 であるとする．この制約に関するラグランジュ乗数 $u \geq 0$ を導入すると，上記問題のラグランジアンは

$$L(u, \mathbf{y}) = f(\mathbf{y}, q) + u \left(\sum_{i=1}^{N} h_i y_i - 1 \right)$$

となる．双対目的関数は

$$L(u) = \max_{\mathbf{y}} L(u, \mathbf{y})$$

であり，双対問題は

$$\min_{u} L(u)$$

となる．双対目的関数を最小化することで，制約を満たす解の最小の上界を求めることができる．劣勾配法を用いて上記の双対問題を解くアルゴリズムを図 1 に示す．図中の $L(u^{(k)}, \mathbf{y})$ を最大化する $\mathbf{y}$ を求める手順は，整数重みのナップサック問題として効率的に解を求めることができる．ここで $\alpha^{(k)}$ は u の更新時のステップ幅を定めるパラメータであり，$\alpha^{(k)} = 0.5/k$ とした．$\lim_{k \to \infty} \alpha^{(k)} = 0$ かつ $\sum_{k=1}^{\infty} \alpha^{(k)} = \infty$ ならば，$k \to \infty$ のときに $L(u)$ がその下限に収束することが知られている [Korte 08]．

4.1 単語の異なり数を加味した制約

ラグランジュ緩和を用いることで，より複雑な形の制約を設けることも可能である．前節で導入した制約

Input: ドキュメント集合 D，クエリ q,
Initialize: $u^{(0)} \leftarrow 0$
for $k \in \{1, \ldots, K\}$ **do**
  $\mathbf{y}^{(k)} \leftarrow \arg\max_{\mathbf{y}} L(u^{(k-1)}, \mathbf{y})$
  $u^{(k)} \leftarrow u^{(k-1)} - \alpha^{(k)}(c_q(\mathbf{y}^{(k)}) - 1)$
  if $u^{(k)} < 0$ **then** $u^{(k)} = 0$
return $\mathbf{y}^K$

図 1: 単語の出現を保証するクエリフォーカス要約のアルゴリズム

は，クエリに含まれる語のうち，いずれかが要約中にが含むことを保証するものであった．この制約は弱いものであるため，より強い制約を加えることで，生成される要約の精度を高めることが期待できる．

より強い制約として，クエリ中に含まれる各語ごとに制約を用意する方法がある．すなわち，クエリ中に M 個の語が含まれていたなら，その各語を含む文の数に関する M 個の制約を用意する．しかし，クエリが質問文のように多くの語からなる場合，制約を満たす解が存在しない可能性が高くなる．そこで，上記二種類の制約の間の強さをもつ制約をラグランジュ緩和によって導入する．つまり，クエリに含まれる語のうち，n 個の異なる語が要約中に出現することを制約として導入する．この制約を素朴に表現すると，$\sum_{t \in q} \sigma(c_t(\mathbf{y})) \geq n$ と書ける．ここで $\sigma(x)$ は，$x \geq 1$ のときに 1，それ以外のときに 0 を返す関数とする．$c_t(\mathbf{y})$ は単語 t を含む文の数を返す関数とする．しかし，この制約のラグランジュ緩和を目的関数に加えると，双対目的関数が y_i の線形和とならなくなるため，ナップサック問題として解くことができない．

ナップサック問題として解くことを可能とするため，クエリ中の n 個の異なる語が要約中に出現するという制約を，複数の線形式に対する制約の組合せとして表現する．例えば，a, b, c をクエリ中の 3 つの語であるとする．これらの語のうち，2 つ以上が要約に出現する制約は，以下の 3 つの制約によって記述できる．

$$\begin{aligned} c_a(\mathbf{y}) + c_b(\mathbf{y}) &\geq 1 \\ c_b(\mathbf{y}) + c_c(\mathbf{y}) &\geq 1 \\ c_c(\mathbf{y}) + c_a(\mathbf{y}) &\geq 1 \end{aligned}$$

このように，M 個の語のうち n 個の異なる語を含む制約は，${}_M C_{M-n+1}$ 個の y_i の線形式である制約によって表現することができる．この方法は，M，n ともに大きな値をとる場合には制約の数が膨大になり，効率的に処理できなくなる可能性がある．しかし，n が小さな場面では十分に実用的である．

TOPIC-ID: 0500
TITLE: クローン羊ドリーに関する記事群
QUESTION:

- 「ドリー」は何の名前か？
- クローン羊ドリーはどこで誕生したか？
- クローン羊ドリーは、胎児細胞ではなく何の複製であることが実験で確認されたか？
- クローン羊ドリーは何からつくり出されたか？
- クローン羊ドリーの元になった雌羊の細胞とドリーの細胞とで、何が同一であると確かめられたか？
- クローン羊ドリーが妊娠中の羊から採った乳腺細胞をもとにつくりだされたことについて、どのような批判があったか？
- クローン羊ドリーの細胞の寿命は普通の羊と比べてどうであることが分かったか？

図 2: TSC3 データセットの質問文の例

4.2 固有表現を含む要約への応用

クエリとして質問文が用いられるような場合には，生成される要約は質問の内容を反映した質問フォーカス要約 [Hirao 01] となるべきであろう．質問文が固有表現について尋ねる内容であるならば，その内容に対応した固有表現を必ず含むようにすれば，要約の精度が向上することが期待できる．

5 実験

5.1 実験設定

TSC(Text Summarization Challenge)3 データセット [Hirao 04] を用いて提案手法の検証を行った．TSC3 データセットはクエリフォーカス要約のためのデータセットであり，30 トピックに関連するドキュメントと質問文，質問文に対する要約からなる．TSC データセットのトピック，質問文の例を図 2 に示す．要約の評価には ROUGE-1[Lin 04] を用いた．TSC3 データセットに含まれる 30 トピックについて ROUGE-1 スコアを計算し，30 トピックの平均を調べた．各文のスコアは，文の tf·idf スコアと，文とクエリとのコサイ

表 1: 制約を用いたときの ROUGE-1 スコア

手法	ROUGE-1
base	0.452
word($n=1$)	0.454
word($n=2$)	0.467
word($n=3$)	0.474

表 2: 固有表現の制約を加えたときのスコア

手法	ROUGE-1
word($n=1$)+ne	0.462
word($n=2$)+ne	0.467
word($n=3$)+ne	0.482

ン類似度の和とした．tf·idf スコアは最大のスコアを 1 として $[0,1]$ の範囲にスケールを揃えたものを用いた．

単語の異なり数を加味した制約として，各質問文から名詞もしくは未知語を抽出し，それらのうち n 個の異なる単語が生成される制約に含まれることを制約とした．$n=1,2,3$ とした．

固有表現を制約に用いる手法についても検証した．要約対象の文の集合に固有表現抽出を適用し，クエリ中の各質問文について，尋ねている内容をもとに，固有表現の出現数に関する制約を定めた．例えば，クエリ中に質問文が 3 文あり，それぞれが場所，人名，日付に関する固有表現を尋ねたものであったとする．このとき，生成された要約中に出現する場所，人名，日付に関する固有表現が，それぞれ一つ以上出現することを制約として加える．ベースラインとして，長さ制約 (2) のもとでスコア関数 (1) を最大化したものを用いた．この最大化は整数重みナップサック問題となっているため，効率的に解を求めることができる．

5.2 結果

ROUGE-1 スコアを表 1 に示す．表中の base はベースライン，word は各質問文の語数を制約として加えた結果である．n はクエリ中の語の出現数の制約のパラメータである．クエリ中の語の出現に関する制約を加えることによって，ROUGE-1 スコアが base を上回ることが確認できた．また，n の値が大きくなるとスコアが改善される様子も確認できた．n を大きくすることは，制約を強くして可能な解を制限することに相当するため，今回の問題設定では設定した制約が ROUGE-1 スコアの上昇に寄与するものであったことが確認できる．

クエリ中の語の制約に加え，固有表現によるスコアを加えたときの ROUGE-1 スコアを表 2 に示す．$n=1,2,3$ のいずれの場合も，固有表現に関する制約を加えることで，ROUGE-1 スコアの上昇が確認できる．

6 おわりに

本稿では，生成される要約中にクエリ中の語が含まれることを保証するクエリフォーカス要約手法を提案した．クエリ中の語を含むことを制約として表現し，ラグランジュ緩和を用いて要約を作成することで，制約を満たす要約を効率的に求めることができる．適切な条件を追加することによって，制約を加えないときと比較して性能が向上することを確認した．

参考文献

[Chang 11] Chang, Y.-W. and Collins, M.: Exact Decoding of Phrase-Based Translation Models through Lagrangian Relaxation, in *Proc. of EMNLP*, pp. 26–37 (2011)

[Daumé 06] Daumé, H. and Marcu, D.: Bayesian Query-Focused Summarization, in *Proc. of ACL* (2006)

[Hirao 01] Hirao, T., Sasaki, Y., and Isozaki, H.: An Extrinsic Evaluation for Question-Biased Text Summarization on QA Tasks, in *Proc. of NAACL workshop on Automatic Summarization* (2001)

[Hirao 04] Hirao, T., Okumura, M., Fukushima, T., and Nanba, H.: Text Summarization Challenge 3, in *Proc. of the NTCIR-4*, pp. 407–411 (2004)

[Koo 10] Koo, T., Rush, A. M., Collins, M., Jaakkola, T., and Sontag, D.: Dual Decomposition for parsing with Non-Projective Head Automata, in *Proc. of EMNLP*, pp. 1288–1298 (2010)

[Korte 08] Korte, B. and Vygen, J.: *Combinatorial Optimization: Theory and Application*, Springer Verlag (2008)

[Lin 04] Lin, C.-Y.: ROUGE: A Package for Automatic Evaluation of Summaries, in *Proc. of Workshop on Text Summarization Branches Out*, pp. 74–81 (2004)

[Rush 10] Rush, A. M., Sontag, D., Collins, M., and Jaakkola, T.: On Dual Decomposition and Linear Programming Relaxations for Natural Language Processing, in *Proc. of EMNLP*, pp. 1–11 (2010)

[Rush 11] Rush, A. M. and Collins, M.: Exact Decoding of Syntactic Translation Models through Lagrangian Relaxation, in *Proc. of ACL/HLT*, pp. 72–82 (2011)

[Tang 09] Tang, J., Yoo, L., and Chen, D.: Multi-topic based Query-oriented Summarization, in *Proc. of SDM*, pp. 1148–1159 (2009)

[Tombros 98] Tombros, A. and Sanderson, M.: Advantages of Query Biased Summaries in Information Retrieval, in *Proc. of SIGIR* (1998)