不要文除去を目的とした重要文抽出システム

天野 禎章　　横山 晶一
山形大学大学院理工学研究科

1. はじめに

我々は、人手作成により近い要約を生成するシステムの構築に取り組んでいる[1]。単一テキスト要約では、重要文抽出した結果を文短縮（重要個所抽出）する構成が一般的である[2]。これは人が要約を作成する際、不要な文に関しては短くまとめて残すのではなく、一文全体を削除するためだと考えられる。この仮説に基づき、本研究では文短縮工程の前段階として、不要文除去を目的とした重要文抽出システムを構築した。

複数テキスト要約においては、高圧縮高精度が求められるが、本システムでは中程度の要約率に関して原文の情報を保った結果の出力を目標として定めた。目標達成には、半分程度の要約率で七割半の評価値を越える必要がある（累積バグや商品需要予測などに用いられる成長曲線に当て嵌めて定めた）。

本システムでは目標達成のため、ベースとなる複数システム（以降、『ベースシステム』と呼称する）から入力文数に応じた複数個の要約集合を作成し、それらを組み合わせた。これは各ベースシステム（PLSI[3]やHITSアルゴリズム[4]など、多言語でも活用可能な情報による要約システム）に内在する信頼性の高い出力を集積することで、優れた結果が得られると考えたためである。

構築したシステムは、F値を用いて評価した。結果は目標の達成にまでは至らなかったものの、作成された複数個の重要文集合には、ベースシステムを越える出力が存在した。

2. 先行研究

本システムのように、複数アルゴリズムの出力結果を組み合わせて要約を行った研究がある[5]。この研究では、Web ページのクラスタリングを目的として、不要な情報の除去に要約を行った。要約作成の工程では、次の5つを利用してベースとする結果を導き出している。

・Luhn が提案した古典的手法[6]（S_{luhn}）
・潜在的意味解析（LSA）[7]（S_{lsa}）
・Web ページの構造情報（S_{cb}）
・HITS アルゴリズム（S_{HITS}）
・Supervised summarization（$S_{\sup}$）

これらを次に示す式1のように、各結果に対して教師なし学習による重み付けを行い（w_1 から w_5）、スコアの高い文を出力している。

$$S = w_1 S_{luhn} + w_2 S_{lsa} + w_3 S_{cb} + w_4 S_{HITS} + w_5 S_{\sup} \quad (1)$$

2.1 PLSI[3]を用いた先行研究

重要文抽出では、各文へのスコア付けがポイントとなる。単純に単語頻度を重要度とするとき、類語や反意語などの意味的繋がりは無視される。この問題を解決するため、シソーラスを利用することが多い。しかしながら、人手によるシソーラス構築はコスト面や新語への対応などの問題点がある。

LSA は、高次元の文書ベクトルを低次元の空間へ射影する手法で、情報検索分野で検索精度の改善に用いられている。これは高次元の空間では別々に扱われていた検索語が、低次元の空間では関連語として扱われる可能性があるためである[8]。この次元削減により、多義性を解決した検索が可能となり、シソーラスを利用せずとも同程度の結果が得られるとされている。

このLSAよりも高い性能を持つとされる確率的潜在意味解析（PLSA）は、次元の圧縮を確率モデルに基づき行う手法である。PLSIを要約システムで用いる有用性は、重要文抽出に利用した研究[9]において、既に述べられている。この研究では、単語頻度行列に対してPLSIを適用する手法（PROC1,PROC3）と、単語頻度行列から文間のコサイン類似度を求め、閾値（0.2）でノードを接続する文書グラフを描写し、PLSIを適用する手法（PROC2,PROC4）で実験している。なお、PROC1 と PROC2では、PLSIを介して算出された確率 $P(d \mid z)$ を文の重要度とし、PROC3 とPROC4 では、次の式2で示すRを文のスコアとして利用している。

$$R = \sum_z P(d \mid z) P(z) = P(d) \quad (2)$$

2.2 ランキングアルゴリズムを用いた先行研究[10]

グラフベースのランキングアルゴリズムでは、各ノードとノード間を繋ぐリンクが与えられたとき、ノード同士を結びつける情報を基に、ノードの重要性をランク付けする。これにより、文書間の参照関係の解析やWebページのリンク構造の解析などが成功している。したがって、文の重要度の算出にランキングアルゴリズムを用いることは有用といえる。

Kleinberg が考案した HITS（Hypertext Induced Topic Selection）[4]は、Web ページのランキングを目的に開発されたアルゴリズムであり、リンク関係からAuthority（被リンク数の多いページ）と Hub（リンク数の多いページ）を求める。この Authority と Hub は、次の式3と式4を反復計算することで算出される。

$$\mathrm{HITS_A}(V_i) = \sum_{V_j \in In(V_i)} \mathrm{HITS_H}(V_j) \quad (3)$$

$$\mathrm{HITS_H(V_i)} = \sum_{V_j \in Out(V_i)} \mathrm{HITS_A(V_j)} \quad (4)$$

HITS アルゴリズムではページの内容を考慮せず、リンク数にのみ着目する。よって、参照数・被参照数が多くなれば、Authority 値と Hub 値は高くなるという問題点が指摘されている。

Page らが提案した PageRank[11]は、Google が採用していたことで有名な Web リンク解析を目的にしたアルゴリズムである。PageRank では、参照と被参照を一つのモデルで、反復して算出する(式5)。

$$\mathrm{PR(V_i)} = (1-d) + d * \sum_{V_j \in In(V_i)} \frac{\mathrm{PR(V_j)}}{|\mathrm{Out(V_j)}|} \quad (5)$$

ダンピング・ファクター(d)は、0から1の間で設定される(通常は 0.85 だが、意図的にランキングを上げようとするページは小さく設定される)。

これらのランキングアルゴリズムを重要文抽出に採用した研究がある[10]。この研究では、文間の類似度を閾値にしたグラフに対して、次の式のように重み付け(重み w_{ij} は V_i と V_j の類似度)し、文番号に着目した有向グラフ(前向き: directed forward と、後ろ向き: directed backward)と、無向グラフ(両方向からの接続)に関してランキングアルゴリズムを適用し、文の重要度を求めている。

$$\mathrm{HITS^w_A(V_i)} = \sum_{V_j \in In(V_i)} \mathrm{w_{ji} HITS^w_H(V_j)} \quad (6)$$

$$\mathrm{HITS^w_H(V_i)} = \sum_{V_j \in Out(V_i)} \mathrm{w_{ij} HITS^w_A(V_j)} \quad (7)$$

$$\mathrm{PR^w(V_i)} = (1-d) + d * \sum_{V_j \in In(V_i)} w_{ji} \frac{\mathrm{PR^w(V_j)}}{\sum_{V_k \in Out(V_j)} \mathrm{w_{kj}}} \quad (8)$$

3. ベースシステム

本システムでは、次の5つから生成される重要文集合をベースシステムの結果とした。

(a) 単語頻度と文書内で最初に出現した位置
(b) 単語頻度ベースのグラフに PLSI を適用
(c) コサイン類似度ベースのグラフに HITS を適用
(d) タイトル文に着目したコサイン類似度ベースのグラフに PageRank を適用
(e) ベースシステムの結果の集約

このうち、(c)と(e)は、それぞれ二つの重要文集合を生成する。よって、計7つがベースシステムとなる。

3.1 単語頻度と文書内で最初に出現した位置

人が文章を読むとき、何かしらの目的がない限り、始めから終わりへと読み進める。よって、出現位置が早い単語ほど読み手の印象に残るため、重要性が高くなると仮定した。この仮定に基づき、各入力文に対して、内容語の頻度と文書内で最初に出現した位置(minPos)を利用して、次の式 9 でスコアリングした。

$$S_{TF*imPOS} = \sum_{l=0}^{m} TF_l \times \frac{N}{\log(\min Pos_l + 1)} \quad (9)$$

パラメータ m は文の内容語数、N は文の総数である。このスコアが高い文を順次出力する。これはTFとLEAD手法の改良として導入した。

3.2 単語頻度行列グラフを用いた PLSI

先行研究では、単語頻度行列にPLSIを適用して求められた確率(EM アルゴリズムで算出した $\mathrm{P(w \mid z)}$、$\mathrm{P(d \mid z)}$、$\mathrm{P(z)}$)からスコア R を計算し、それを用いた PROC3 での結果が優れていた。これを日本文に対して確認したところ、同様の結果が得られた。よって、本システムでもPROC3を用いた。

3.3 コサイン類似度ベースの HITS アルゴリズム

より多い文と関連性のある文は重要であると考えられるため、Authority 値と Hub 値を重要文抽出で利用するのは有効である。しかしながら、類似度を重みとして加えると、類似度の影響が多くなる可能性がある。

よって、本研究では、単語頻度行列からコサイン類似度を求め文書グラフを描写し、類似度重みのない式3と式4を採用した。このとき、先行研究で評価値が高くなった前向きグラフ(文番号1から文番号2や文番号3などへ接続させるグラフ)に対して得られた Hub 値を文のスコアとした場合と、同じく高くなった後ろ向きグラフ(文番号3から文番号1や文番号2などへ接続させるグラフ)に対して得られた Authority 値を文のスコアとした場合、この二つを独立に用いて重要文抽出した(他の組み合わせでも試したが、先行研究通りの結果だった)。

3.4 タイトル文ベースの PageRank アルゴリズム

タイトル文を根として、類似度が閾値よりも高い文へ順次接続したグラフに PageRank アルゴリズムを適用した。この結果に対して、各文の TF 値をかけあわせた値を文のスコアとして、高い文から選択する。

3.5 ベースシステムの結果の集約

複数の情報を用いて文のスコアを算出する場合は、先行研究のように、重み付き総和が用いられる。このとき、スコアの重みは確率や機械学習などで決定されることが多く、最終的な出力に大きく依存する。よって本システムでは、重み付き総和による文のスコアもベースシステムとして取り入れ、最終的な出力には異なる手法を用いた。

重み付き総和による文のスコアを用いたベースシステムでは、各システムの重み(w)とシステム評価の尺度で

あるF値(NTCIR[12]のdryrunデータで評価した結果)を、12ドメイン[13]の頻度を素性データにTinySVM[14]で学習し、信頼度曲線に近似させて定めた。このとき、各ベースシステムで算出したスコアが平均的に高い文が集まると期待される。

また、同要約率時出力における各ベースシステムが算出した文のスコア(最大が1となるように調整)が高い文から構成される重要文集合も採用した。これにより、各ベースシステムで特徴付けられる文の集約が期待される。

4. 複数要約率出力結果の組み合わせ

本システムでは、入力文のうち平均文字数を超える文数から1を引いた値と、ベースシステムの総数を掛け合わせた数だけの要約が生成される。

例えば、平均文字数を越える文が10個でベースシステムが7個の場合、要約率 10%の出力を7個、要約率20%の出力を7個と続き、要約率 90%の出力が7個の合計63個の重要文集合を生成する。

これは各ベースシステムから要約率を変化させた複数個の出力集合を組み合わせることで、単純にユーザーが指定した要約率での出力だけを作成するときよりも優れた結果を導き出せると考えたためである。

上記の仮定を基に、各ベースシステムから作成された重要文集合を組み合わせて最終的な出力を求める。組み合わせ方法は、各ベースシステムの出力集合から一つずつ選択する(上記の例では、$9^7 = 4782969$ 通りとなる)。このとき、各ベースシステムが出力した重要文集合からより多くの集合に属する文(今回は最大で7)を、次の式で定義するスコアが高い順に、ユーザーが指定した要約率を満たすまで抜き出す。

$$S'_n = w_n S_n * \log\left(\frac{100}{M} + 1\right) \quad (10)$$

Mは文番号nが出力する最も低い要約率であり、S_nは出力したベースシステムがその要約率時に付与したスコアであり、そのときの重みをw_nとする。これは要約率がより低いときの結果の方が、要約率が高いときの結果よりも重要であるという仮定に基づくものである。

こうして得られた重要文集合は、より多くのベースシステムのより重要な結果が集約したと考えられる。

5. 評価結果

作成したシステムを NTCIR[12]で提供されている正解データ(Formal Run)を用いて評価した。評価尺度には精度と再現率、F値を用いている[1](式11、式12、式13)。求められた F 値が次の表1である。ベースシステムの各評価結果をベースラインとしてともに提示している。

$$\text{精度} = \frac{\text{システムが選んだ正解 文の数}}{\text{システムが選んだ文の 総数}} \quad (11)$$

表 1　システムの F 値

	CR10%	CR30%	CR50%
TF*imPOS	0.308	0.436	0.577
PLSI	0.334	0.446	0.602
HIT-AUTH	0.510	0.496	0.593
HIT-HUB	0.509	0.499	0.605
TF*PGR	0.358	0.464	0.591
CB-SCR	0.426	0.484	0.570
CB-MAX	0.397	0.464	0.601
SYSTEM-max	**0.598**	**0.625**	**0.706**
SYSTEM-ave	**0.315**	**0.429**	**0.541**

$$\text{再現率} = \frac{\text{システムが選んだ正解 文の数}}{\text{人間が選んだ正解文の 総数}} \quad (12)$$

$$F\text{値} = \frac{2 \times \text{再現率} \times \text{精度}}{\text{再現率} + \text{精度}} \quad (13)$$

TF*imPOS は(a)、PLSI は(b)、HIT-AUTH は(c)の後ろ向きグラフの Authority 値で、HIT-HUB は前向きグラフの Hub 値、TF*PGR は(d)であり、CB-SCR は重み付き総和を用いた結果、CB-MAX は各ベースシステムの最大スコアの集約を示している。SYSTEM-max は、本システムで出力した複数個の重要文集合を評価したとき最大となった結果を集めた値であり、SYSTEM-ave は全重要文集合の平均評価値である。

重要文集合の最大評価値では、各ベースシステムを上回ったが、平均値ではほぼ全てのベースシステムよりも低くなった。これは出力重要文集合内に好ましくない結果が含まれており、それが評価を下げているためである。よって、最終的な出力の決定方法やシステムの組み合わせ方法などに改良が必要である。

また、重み付き総和の結果がベースシステムよりも低い。最適な重みなら、これらよりも優れた評価となると予想される。よって、設定方法の修正も必要である。

6. 追加実験

本システムでは、各ベースシステムが作成する要約数を、平均文字数を越える文の数とした。これにより、生成する重要文集合は多様化し、組み合わせ結果に大きく影響を与えた。しかしながら、文数が増えるにつれ、計算量は増大する。そこで作成する要約数と評価結果の関係から、最適と思われる数について調べた。

実験では、ベースシステムが作成する要約数を予め設定しておき、これを変化させた。得られた出力のF値を最大と平均値ごとにまとめたのが次の図である。要約数の増加とともに最大評価値が上昇しているが、作成数6付近で横這いとなっている。したがって、ベースシステムが作成する要約数の基準は6が妥当と思われる(この場合の組み合わせ数は$6^7 = 279936$ 通りとなる)。

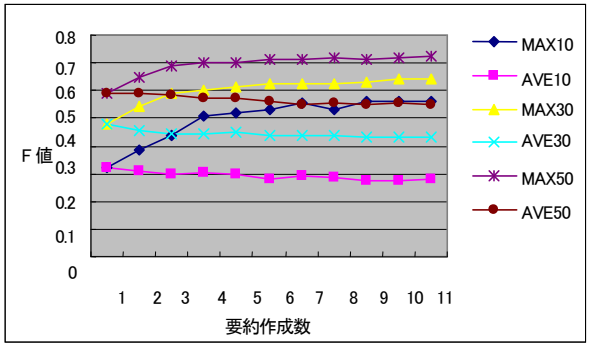

図　要約作成数と F 値

7. 今後の課題

本稿では、文短縮工程の前段階としての不要文削除を目的にした重要文抽出システムを構築した。システムでは、設定した目標を達成するために、複数のベースシステムから複数要約率での重要文集合を作成し、それらを組み合わせたが結果は芳しくなかった。

しかしながら、評価の最大値と平均値にまだ差がある。よって改良次第では、目標の達成が期待される。

異なるアルゴリズムによるベースシステムの追加は、より多様な要約集合の作成が可能になり、評価の向上が期待できる。ＰＬＳＩを改良したＬＤＡ（Latent Dirichlet allocation）を文章要約に適用した研究[15]があり、これの導入も考えている。

先述した通り、ベースシステムの重み設定が充分ではない。頻度や出現位置などの7つの特徴を素性に用いてロジスティック回帰分析から重みを求めた研究[16]があり、導入による改善が期待される。

システムの組み合わせ方法の改良もまた、大きな課題点である。組み合わせるパターンを遺伝子列に見立てて遺伝的アルゴリズムを利用したり、動的計画法や貪欲アルゴリズムなどで組み合わせの選択を行ったりしたが、望ましい結果は得られなかった。複数テキスト要約の手法や、導入が可能であろう組み合わせ最適化を実験し、計算量の削減とシステム評価の向上を狙う。

謝辞

NTCIR より提供されるテストコレクションがシステムの構築に大きく貢献しました。NTCIR に多大な感謝を申し上げます。

参考文献

[1]天野禎章, 横山晶一, 橋本力, “主題・焦点に基づく文統合工程を実装した要約システム”, 言語処理学会第14回年次大会, Mar. 2008

[2]奥村学, 難波英嗣, “テキスト自動要約”, オーム社(2005)

[3]T. Hofmann, ” Probabilistic Latent Semantic Indexing” Proceedings of the Twenty-Second Annual International SIGIR Conference on Research and Development in Information Retrieval, 1999

[4]Kleinberg, J.M. “Authoritative sources in a hyperlinked environment”, Journal of the ACM 46(5), pp.604-632.

[5]Dou Shen ,Qiang Yang, Zheng Chen ,”Noise reduction through summarization for Web-page classification”, Information Processing and Management 43, pp.1735–1747 (2007)

[6]H.P.Luhn, “The automatic creation of literature abstracts”, IBM Journal of Research and Development, Vol.2, No.2, pp.159-165(1958)

[7]S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, R. Harshman: Indexing by Latent Semantic Analysis. Journal of the Society for Information Science, 41(6), 391-407, 1990.

[8]北研二, 津田和彦, 獅々堀正幹, “情報検索アルゴリズム”, 共立出版(2002)

[9] Harendra Bhandari, Masashi Shimbo, Takahiko Ito, Yuji Matsumoto, “Generic Text Summarization Using Probabilistic Latent Semantic Indexing”, Proceedings of the Third International Joint Conference on Natural Language Processing, pp.134-140, 2008

[10]Rada Mihalcea, “Graph-based Ranking Algorithms for Sentence Extraction, Applied to Text Tummarization”, Proceedings of the ACL 2004 on Interactive poster and demonstration sessions, July.2004

[11] Sergey Brin and Lawrence Page, “The anatomy of a large-scale hypertextual Web search engine”, Computer Networks and ISDN Systems, 33(1) pp.107-117, 1998.

[12]NTCIR, “http://research.nii.ac.jp/ntcir/index-ja.html”（アクセス日時:2008.12.17）

[13]橋本力, 黒橋禎夫, “基本語ドメイン辞書の構築と未知語ドメイン推定を用いたブログ自動分類法への応用”, 自然言語処理, Vol15.No5, pp.73-98, 2008

[14]奈良先端科学技術大学,「TinySVM」

[15] Rachit Arora, Balaraman Ravindran, ”Latent dirichlet allocation based multi-document summarization”,Proceedings of the second workshop on Analytics for noisy unstructured text data, pp.91-97, July.2008

[16]Wen-tau Yih. Joshua Goodman. Lucy Vanderwende. Hisami Suzuki, ”Multi-Document Summarization by Maximizing Informative Content-Word”,In Proc. of the 20th International Joint Conference on Artificial Intelligence, pp.1776-1782, Jan.2007