ユーザとのインタラクションを導入した複数文書要約システム

酒井 浩之　　増山 繁
sakai@smlab.tutkie.tut.ac.jp,　masuyama@tutkie.tut.ac.jp
豊橋技術科学大学 知識情報工学系

1　はじめに

近年のコンピュータと情報通信技術の進歩により，多くの情報を得ることができるようになった．しかし，それは情報の氾濫を招き，そのことにより，文書情報を要約するテキスト自動要約の必要性が高まってきている．その中でも，複数の文書を自動要約する研究は，検索結果等の大量の文書群を大幅に縮めて１つの文書として提供することにより迅速な理解を可能にする．そのため，その技術の確立が強く望まれており，近年，盛んに研究されている [3]．複数文書要約とは，検索質問に適合する文書集合から, 指定された長さの要約を作成するタスクと定義される．ところが，検索質問に適合する複数の文書を要約するとはいえ，文書群中や文書中の話題は一様ではない．例えば，ソニーのＡＩＢＯ発売に関する文書群には，ＡＩＢＯの動作に関するものや販売方法に関するものなど，様々な話題が含まれている．要約された文書は，これらの話題を全て盛り込んでいることが望ましいが，少ない文字数では，すべて表現するのは困難である．そこで，多様な要約要求に対応するために，ユーザとのインタラクションを導入した複数文書要約システムを開発する．まず，要約対象となる文書群から，検索質問と関連したキーワードを自動的に抽出し，ユーザに提示する．次にユーザが要約要求に適したキーワードを選択，もしくは，適さないキーワードを削除し，選択したキーワードによって生成される要約文書を制御する．

本要約システムは，検索と要約の評価のためのワークショップ NTCIR4 [1] における要約タスク (以降，TSC3) に参加した．ただし，参加の際は，本稿で紹介するシステムにおいて，ユーザとのインタラクションを一切行わず，全自動で要約を生成した．(後述の質問集合も使用していない．) 本稿では，本要約システムの手法と，TCS3 の抜粋作成タスクの評価結果について述べる．

2　関連キーワードの抽出

要約対象の文書集合 S から検索質問と関連したキーワード (以降，関連キーワードとする)t_i $(i = 1, 2, \ldots, m)$ を抽出する手法について述べる．要約対象となる複数文書から関連キーワードを抽出する手法は，複数文書中の名詞にスコア付けを行い，スコアが高い順に抽出する．名詞のスコアは，tf・idf 法 [1] をベースに，次の 5 つの条件をより満たすほど高い値を割り当てる．

1. 要約対象の文書集合 S に多く出現する名詞.
2. S の各文書に均一に出現する名詞.
3. 文書の最初（第 1 文）に出現する名詞.
4. S を時系列に従って並べた場合に，最初（第 1 記事，第 2 記事...）に出現する名詞.
5. S 以外に出現している頻度が小さい名詞.

本要約システムにおける関連キーワードの抽出手法は，以下に示す 3 つのステップに分けられる．

Step 1: 文書集合 S に含まれる名詞 t_i $(i = 1, 2, \ldots, n)$ を抽出する．

Step 2: t_i $(i = 1, 2, \ldots, n)$ に対して，計算式 (1) で重み $W(t_i, S)$ を計算する．

Step 3: 重み $W(t_i, S)$ の上位 α 個を関連キーワードとして抽出する．□

Step 2 における計算式 (1) を以下に示す.

$$\begin{aligned} W(t_i, S) \;=\; & (0.5 + \frac{Tf(t_i, S)}{max_{i=1,\ldots,n} Tf(t_i, S)}) \\ & \times (0.5 + \frac{En(t_i, S)}{max_{i=1,\ldots,n} En(t_i, S)}) \\ & \times max_{s \in S} \frac{1 + nl(s) - nlf(t_i, s)}{nl(s)} \\ & \times max_{s \in S} \frac{1 + |S| - rt(t_i, s)}{|S|} \\ & \times \log \frac{|N|}{df(t_i, N)}, \end{aligned} \tag{1}$$

但し，

$Tf(t_i, S)$**:** 文書集合 S における名詞 t_i の出現頻度．詳細は後述する．

$En(t_i, S)$**:** 文書集合 S において，名詞 t_i の出現確率に基づくエントロピー．詳細は後述する．

$nl(s)$**:** 文書 $s \in S$ の行数,

$nlf(t_i, s)$**:** 文書 $s \in S$ において，名詞 t_i が最初に出現する行番号,

$rt(t_i, s)$**:** 文書集合 S を時系列に従って並べた場合の名詞 t_i を含む文書 $s \in S$ の文書番号,

$df(t_i, N)$**:** 全文書集合 N(すなわち，$S \subset N$) において，名詞 t_i を含んでいる文書の頻度,

$Tf(t_i, S)$ によって，文書集合 S において，高い出現頻度をもつ名詞は，大きい重みが割り当てられる．$Tf(t_i, S)$ は，以下の計算式 (2) で求める．

$$Tf(t_i, S) = \sum_{s \in S} tf(t_i, s), \tag{2}$$

但し，$tf(t_i, s)$ は文書 $s \in S$ における名詞 t_i の出現頻度である．

$En(t_i, S)$ は，文書集合 S において，名詞 t_i の出現確率に基づくエントロピーである．例えば，一つの文書

[1] http://research.nii.ac.jp/ntcir/workshop/work-ja.html

にのみ多く出現する名詞の $En(t_i, S)$ は小さくなる．そのような名詞は，その文書においては重要な名詞であるかもしれないが，検索質問と関連のある名詞ではない可能性がある．逆に，文書集合 S の各文書に均一に出現する名詞の $En(t_i, S)$ は大きくなる．$En(t_i, S)$ は，以下の計算式 (3) で求める．

$$En(t_i, S) = -\sum_{s \in S} P(t_i, s) \log_2(P(t_i, s)), \quad (3)$$

$$P(t_i, s) = \frac{tf(t_i, s)}{Tf(t_i, S)}, \quad (4)$$

式 (1) の第 3 項は，文書の最初の方に出現した名詞に対して高い重みを与えるための項である．これは，文書 (特に，新聞記事) の始めの文には重要な情報が出現していることが多いために導入した．式 (1) の第 4 項は，文書を時系列に従って並べた場合 (新聞記事の場合，発行日順)，最初の文書中における名詞に高い重みを与える項である．この項で，ある事柄に関して時系列順で最初の記事に含まれる名詞ほど高い重みが与えられる．第 3 項，第 4 項の導入の理由は，複数文書要約において，事柄を伝える最初の記事の第 1 文が重要であることが多いからである．式 (1) の第 5 項は，コーパス N における idf 値である．

3 重要文抽出

本要約システムにおける重要文抽出は，ユーザによって選択された関連キーワードと各文との類似度を測り，類似度が高い順に文を抽出する．具体的には，選択された関連キーワードの単語ベクトルと，文の単語ベクトルとの余弦値を測り，余弦の大きい文を重要文として抽出する．つまり，選択されたキーワードを全て含む文が最も重要な文であると認識される．よって，提示されたキーワードをユーザが確認し，必要なキーワードを選択し，不必要なキーワードを排除すれば，重要文として認識される文も変化する．重要文抽出の手法を以下に示す．ここで，提示されたキーワード集合を K，ユーザが選択したキーワード集合を U とする．

Step 1: 関連キーワード t_i $(i = 1, 2, \ldots, n)$ の重みを以下の式で再計算する．ここで α は，キーワードの提示数である．

$$W'(t_i, S) = \begin{cases} (1 + 0.5\alpha) W(t_i, S), & t_i \in U \\ W(t_i, S), & \text{otherwise} \end{cases} \quad (5)$$

Step 2: キーワード集合 K に対して，再計算した重み $W'(t_i, S)$, $t_i \in K$ を要素としたベクトル V_K を生成する．

Step 3: 文 $s \in S$ に対して，文 s に含まれる名詞 t_j の重み $W'(t_j, S)$ を要素としたベクトル V_s を生成する．

Step 4: V_K と V_s の余弦値を求め，類似度 $sim(s, K)$ とする．そして，類似度の大きい順に m 個の文を重要文として抽出する． □

4 冗長な情報の削除

複数文書要約では，複数文書間にまたがる冗長な情報を削除する必要がある．本要約システムでは，まず文ごとの冗長性を測定し，重要文の中で冗長な文を削除する．その後，文書の冗長性を測定し，内容の類似している文書を削除する．まず，冗長な文の削除手法を以下に示す．

Step 1: 文 s_1 と文 s_2 の，各類似度の差 $d(s_1, s_2)$ を求める．

$$d(s_1, s_2) = |sim(s_1, K) - sim(s_2, K)| \quad (6)$$

Step 2: $d(s_1, s_2)$ が極めて小さい値であるなら，$sim(s_1, K)$ と $sim(s_2, K)$ において，時系列順で後の記事に含まれている文を削除する． □

本要約システムでは，Step 2 の閾値を 0.0001 とした．

次に，冗長な文書間の削除を行う．ここで，削除する単位を要約文書 sd_i とし，sd_i とは，文書 d_i から，重要文として抽出された文の集合とする．つまり，要約文書 sd_i は，文書 d_i の抜粋である．以下に手法を示す．

Step 1: 要約文書 sd_1 に対して，要約文書 sd_1 に含まれる名詞 t_n の重み $W'(t_n, S)$ を要素としたベクトル V_{sd_1} を生成する．

Step 2: 要約文書 sd_2 に対して，要約文書 sd_2 に含まれる名詞 t_m の重み $W'(t_m, S)$ を要素としたベクトル V_{sd_2} を生成する．

Step 3: V_{sd_1} と V_{sd_2} の余弦を求め，sd_1 と sd_2 の類似度とする．

Step 4: sd_1 と sd_2 の類似度がある閾値より大きい場合，式 (7) で計算される W_{sd_1} と W_{sd_2} のうち，小さい W_{sd_i} が割り当てられる文書を削除する．

$$W_{sd_i} = \sum_{s \in sd_i} sim(s, K) \quad (7)$$

この処理を，複数文書において，同じ日時，および，1 日違いの要約文書対に対して行う．なお，冗長な要約文書が削除されると，その文書から抽出された文は抽出されなくなるので，抽出される文が変化する．そのため，もし要約文書が削除されれば，残りの文書群に対して重要文抽出を行う．その結果，生成された要約文書で，冗長な要約文書削除を行う．この繰り返しを，要約文書が削除されなくなるまで行う．なお，Step 4 の閾値は 0.85 とした．

5 文中の不要個所削除

文中の不要個所削除は，動詞連体修飾節の削除と接続詞の削除によって行う．ここで，動詞連体修飾節 (以降，動詞連体とする．) とは，名詞 n を修飾している連体修飾節の中で，動詞を含み，かつ，その動詞が名詞を修飾している連体修飾節 $VP(n)$ と定義する．例えば，「ソニーが開発したアイボ」の「ソニーが開発した」は名詞「アイボ」を修飾している動詞連体である．動詞連体修飾節の削除は文献 [4] に基づき，複数文書要約に特化して，より多くの動詞連体修飾節を削除できるように改良した手法である．以下に手法を簡単に示す．

Step 1: 動詞連体 $VP(n)$ に対して，計算式 (8) で重み $W(VP(n), S)$ を算出する．

Step 2: 式 (10) で示される $endf(n)$ が，ある閾値より低い名詞を修飾しており，かつ，重み $W(VP(n), S)$ が，ある閾値より低い動詞連体を省略可能と認定する． □

Step 2 における $endf(n)$ と $W(VP(n), S)$ の閾値は，他コーパスにおいて実験した場合の最適値である 0.7 と 8.7 とした．Step 1 における計算式を以下に示す．

$$W(VP(n), S) = \frac{NM(n)IM(VP(n), S)}{0.5 + 0.5CV(n, S)} \quad (8)$$

但し，

$NM(n)$: 名詞 n における動詞連体の修飾必要性を表す指標 [4],

$IM(VP(n))$: 動詞連体 $VP(n)$ における内容の重要度を推定する項,

$CV(n, S)$: 時系列順に並べた複数文書群 S において，対象としている動詞連体 $VP(n)$ までに，名詞 n が動詞連体によって修飾された回数,

$NM(n)$ は，以下の計算式で計算される．

$$NM(n) = 0.5 + \frac{endf(n)}{J(n)} \quad (9)$$

$$endf(n) = \frac{1 + H(n)}{idf(n)} \quad (10)$$

但し，

$H(n)$: 名詞 n に対して，ある動詞連体が修飾する確率に基づくエントロピー [4].

$idf(n)$: 名詞 n のコーパス N における idf 値.

$J(n)$: 名詞 n が複合名詞であった場合，それを構成している普通名詞の数,

$IM(VP(n))$ は，以下の計算式で計算される．

$$IM(VP(n), S) = 0.5 + R \sum_{c \in VP(n)} I(c, S) \quad (11)$$

$$I(c, S) = \frac{W'(c, S)}{0.5 + 0.5CT(c, S)} \quad (12)$$

但し，

R: 対象としている動詞連体 $VP(n)$ を構成する文節の数,

$W'(c, S)$: 動詞連体 $VP(n)$ に含まれる名詞 c の，式 (5) で計算された重み,

$CT(c, S)$: 時系列順に並べた複数文書群 S において，対象としている動詞連体 $VP(n)$ の出現までに，名詞 c が動詞連体に含まれていた回数,

文献 [4] と大きく異なる点は，複数文書群において，動詞連体によって多く修飾されている名詞を修飾している動詞連体であるならば，省略可能と認定されやすくなることである．これは，複数の文書において，同じ名詞に対して同じ内容の動詞連体が修飾していることが多いからである．

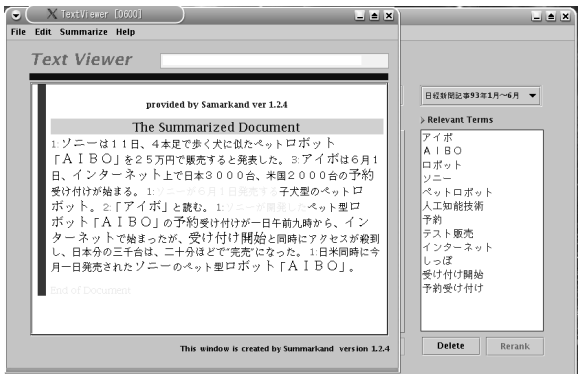

図 1: 関連キーワードと生成された要約文書

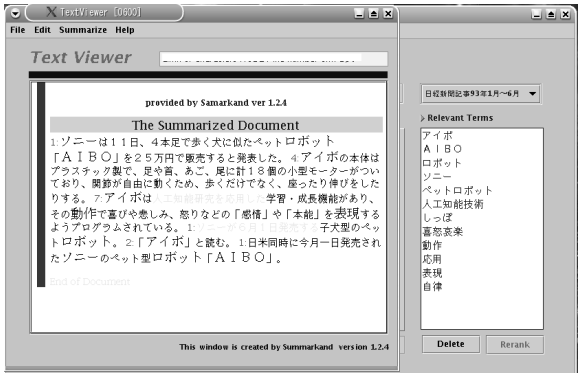

図 2: 関連キーワードを変更して生成された要約文書

6 手法の実装

本要約システムは Java と Perl で実装し，形態素解析器として JUMAN [2]，構文解析器として KNP [3]を使用した．図 1 は，TSC3 の要約課題「アイボ発売に関する記事群を要約せよ」で提示された関連キーワードと，提示された関連キーワードから生成された要約文書を示す．図 2 は，提示された関連キーワードを変更し，アイボの動作や性能に関するキーワードを選択した場合に生成された要約文書を示す．これらの例では，9 文書を 236 文字以内に要約した．図 1 と図 2 を比べると生成された要約文書が異なっており，選択した関連キーワードによって生成される要約文書を制御することができる．

7 TSC3 における評価

本要約システムの評価は，TSC3 における抜粋作成タスクに基づいて行った．TSC3 の抜粋作成タスクは，複数文書からの重要文抽出，および，冗長な情報の削除を評価するタスクである．評価は，人間が抽出した文を正解データとして，TSC3 で定義された Precision と Coverage で行う．結果は Precision と Coverage の調和平均で示す．TSC3 抜粋作成タスクでは，以下の 3 つの情報が与えられる．

1. 要約課題 (例：アイボ発売に関する記事群を要約せよ)

[2]http://www.kc.t.u-tokyo.ac.jp/nl-resource/juman.html
[3]http://www.kc.t.u-tokyo.ac.jp/nl-resource/knp.html

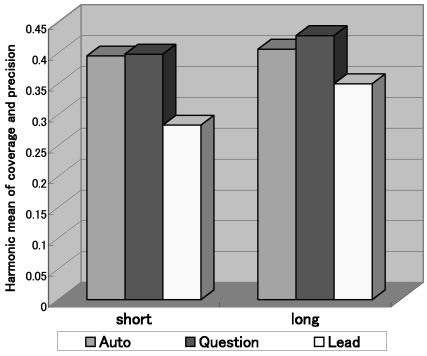

図 3: TSC3 抜粋タスクの評価結果

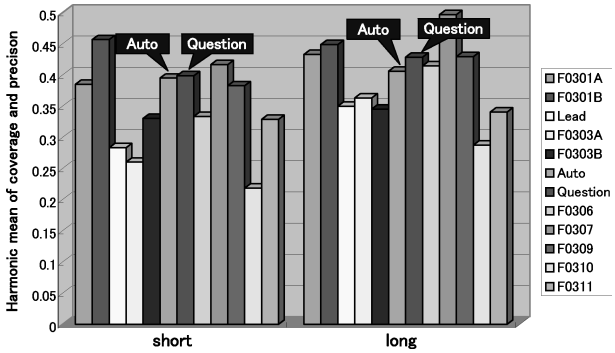

図 4: TSC3 参加システムとの比較

2. 文書集合 (毎日新聞記事 98 年 99 年，および，読売新聞記事 98 年 99 年)
3. 複数文書群中の主要な情報に関する質問集合 (例：ＡＩＢＯの発売開始はいつか？...etc.)

評価において，比較のために以下の 3 つの手法を定義する．

Auto: 関連キーワードを 12 個提示し，全て自動で選択する手法

Ouestion: 関連キーワードから，質問集合に含まれている名詞を選択し，それ以外を削除する手法

Lead method: 指定された文数だけ文書の最初から選択するベースライン手法

Auto は，ユーザとのインタラクションをなくし，自動的に要約を生成する手法であり，TSC3 に参加した手法である．Question は，質問集合をユーザの要約要求とみなし，それに含まれる名詞を選択し，それ以外を削除することで，ユーザとのインタラクションをシミュレートした手法である．結果を図 3 に示す．図 4 に，他の TSC3 参加システムと，我々の手法における Auto を比較した結果を示す．参考のため，Question の結果も併記する．

8 考察

図 4 より，TSC3 参加手法 (Auto) では，short において Precision と Coverage の調和平均が 11 システム中 3 位 (Question は除く) であり，良好な結果を得た．(ただし，Auto は質問集合を使用しておらず，自動で要約文書が生成される．質問集合を使用したシステムと使用していないシステムを順位で比較することはできないので，より高い順位である可能性もある．) また，図 3 より，質問集合を使用してユーザのインタラクションをシミュレートした手法は，より高い評価を得ることができた．そして，short よりも long の方がインタラクションによる評価の向上が顕著である．これは，short は抽出できる文数が少ないため，質問集合によって関連語が変化したとしても，short では抽出される文があまり変化しない．逆に，long では抽出される文が多く変化し，より要約要求に適した文が多く含まれる文集合になるからであると考える．

本要約システムでは提示する関連キーワード同士の類似性判別を行っていない．そのため，例えば「アイボ」と「AIBO」が異なる名詞として認識されている．ユーザに提示する際に，同義語はまとめて提示した方がユーザの負担が減るうえに性能の向上も期待できる．そのため，文献 [2] や，文献 [5] などの手法を用いて，同義語を精度よくコーパスから抽出し，ユーザに提示する際に同義語をまとめて提示する必要がある．

9 結び

ユーザとのインタラクションを導入した複数文書要約システムを開発した．まず，要約対象となる文書群から，検索質問に関連したキーワードを自動で抽出し，ユーザに提示する．次にユーザが要約要求に適したキーワードを選択，もしくは，適さないキーワードを削除し，選択したキーワードによって生成される要約文書を制御する．本要約システムを評価したところ，ユーザとのインタラクションを導入することで評価が向上し，有効性を確認した．

謝辞

本研究は文部科学省２１世紀 COE プログラム「インテリジェント　ヒューマンセンシング」，及び，日本学術振興会科学研究費基盤研究 (C)(2)13680444 の援助により行われた．

参考文献

[1] R. Baeza-Yates, B. Ribeiro-Neto, “Modern Information Retrieval,” ADDISON WESLEY, 1999.

[2] T. Hisamitsu, Y. Niwa, “Extracting useful terms from parenthetical expressions by combining simple rules and statistical measures: A comparative evaluation of bigram statistics,” Recent Advances in Computational Terminology, Edited by D. Bourigault, C. Jacquemin, M. -C. L’Homme, John Benjamins Publishing Company, pp. 209-224, 2001.

[3] I. Mani, “Automatic Summarization,” John Benjamins Publishing Company, 2001.

[4] H. Sakai, S. Masuyama, “Unsupervised knowledge acquisition about the deletion possibility of adnominal verb phrases,” In Proc. of Workshop on Multilingual Summarization and Question Answering 2002 (post-conference workshop of COLING-2002), pp.49–56, Taipei, Taiwan, Sep. 2002.

[5] 酒井浩之，増山繁，“コーパスからの名詞と略語の対応関係の自動獲得,” 言語処理学会第 9 回年次大会発表論文集，pp. 226–229，2003.