

3

携帯端末向けコンテンツ変換と自然言語処理

中川 裕志

東京大学情報基盤センター nakagawa@dl.itc.u-tokyo.ac.jp

渡部 聡彦

東京大学情報基盤センター nabe@lib.u-tokyo.ac.jp

携帯環境の現状

すでに携帯電話の加入者数が固定電話の加入者数を上回っている。また欧米では携帯電話は日本ほどではないらしいがPDAがよく使われており、PDAにおける表示技術の研究が盛んに行われている。このような状況を見ると携帯電話やPDAなどの携帯端末を念頭においた情報処理の研究は今後、ますます重要になると思われる。

携帯端末に表示されるコンテンツはテキスト、図形(地図)、静止画、動画、音声(たとえばVoiceXMLなどは注目に値する)と多様化しているが、その基本にはテキストがある。よって、携帯端末へのテキスト表示は実用的価値の高い研究テーマである。テキスト表示を念頭においた自然言語処理という概念は従来あまり多く論じられていない。自然言語処理研究においてはテキスト処理の結果は、構文木、意味表現、用語、Named Entity、チャンク、そしてテキスト、さらには音声、画像という形態をとるが、音声、画像を除けば、こと表示に関する限り通常のパソコンやサーバに接続される1000×1000画素以上の大画面を想定していた。しかし、後に述べるように種々の制約がある携帯端末の画面での表示を想定すると、それなりの自然言語処理技術があつて然るべきである。

この特集テーマの自動要約は、「携帯端末の小さな画面に表示するテキストは短く、理解しやすく、さらに読むためのインタフェースのよいものでなければならない」という直感に直接的に関係する技術である。テキス

ト自動要約はテキスト量の削減ないし滑らかな要約文への変換が中心課題であり、表示を意識した研究は二次の感が否めなかった。例外としては、字幕放送回向けのテキスト作成があり、「短く、分かりやすい」テキストという点では携帯端末表示の場合と相通ずるものがあるが、これは放送媒体を対象にしているので、細かい技術としては異なる点が多い。

このような背景の下に、この解説では携帯端末への表示を想定した自然言語処理技術の中心であるテキスト自動要約、およびそれに付随するいくつかの問題について述べる。

では、従来のテキスト自動要約と携帯端末表示における要約との相違は何であろうか？ これは、次の4つの側面から論じると分かりやすいだろう。すなわち、ハードと通信の環境、利用局面、目的、応用である。

◎ハードと通信の環境

まず画面サイズについて考える。通常のパソコンやサーバの大画面と異なり、携帯電話の画面は8×6文字から10×10文字程度の表示能力である。PDAは画面がもう少し広いが、20×10文字程度の表示が多い。高解像度化は可能だが、携帯端末であることから画面の物理的サイズには限界がある。人間が読むということを考慮すれば、表示文字数はPDA程度に限られてくるであろう。したがって、Webページのコンテンツを直接携帯電話やPDAに表示しようとしても可読性の低いものになってしまう。一例として、通常のWebページの表データを携帯端末に直接表示した例を示す。可読性が悪く、理

郵便料金表 通常郵便物	
便物 (認 可を 受け た定 期刊 行物 ・開 封)	寄人から差 し出される もの 50g まで どこに どこに 50g まで 毎月 3回 以上 発行 する まで 5 kg まで 5
	8円
	3円増

図-1 表データを PDA に表示した例

解しにくい表示である(図-1)。

一方、携帯端末における処理能力は大きく向上しており、メモリ量は10MB程度、CPU速度は数10MHzになっており、かなりの処理能力がある。しかし、単独で自然言語処理の重い処理をするのはまだ無理であろう。携帯端末上ですべての処理を行うのが無理だとすれば、携帯端末へのコンテンツ配信システム全体のアーキテクチャも検討課題になってくる。

通信は無線回線の高速化が進み384KB/sとブロードバンド化している。さらにもっと高速な無線LANのカバー範囲が広がる傾向もある。こうしてみると、テキストを扱っている限り通信速度は大きな問題にはならない。

◎利用局面

携帯端末は移動中に使うことが多い。たとえば、乗り物の待ち時間などに使うことが多い。乗車中に携帯電話でメールを読んでいる人も多いが、下車駅が近づくとき携帯電話を切る。このように、携帯端末の利用はかなりの場合、時間的に制限された状況、すなわち時間圧の高い状況で行われる。

◎目的

画面が小さいこと、使用時の高い時間圧を念頭におくと、少ない表示量でできるだけ多くの情報を理解しやすいかたちで伝えることが表示のためのテキストの要件である。この要件を満たすテキストを既存のテキストから変換して生成することが目的になる。

一方で、テキストを表示する際のインタフェースも重要である。たとえば、スクロールして長いテキストを読むためにはクリック回数が増加し必ずしも扱いやしくない。もちろん、1画面で1文書が理想だが、これも困難な場合が多く、いろいろな対策が提案されている。

結局、目的を達するにはテキスト自動要約単独ではなく、総合的なインタフェース技術として捉える必要

がある。ハイパーテキスト化や図表の扱いも要検討であろう。

◎応用

簡潔で理解しやすいテキストを目的とした要約は、高齢者や幼児向けのデジタルディバイド対策としても期待ができる。

以下、本稿では、まずコンテンツのPDA端末表示ブラウザ、次に携帯端末向けの自動要約技術、要約の一種である縮約、展望を述べる。

インタラクティブなブラウザ

携帯端末へのコンテンツ表示においては、インタラクティブ性を勘案した可読性についての考察や、小画面を活かしたブラウザが大切である。この章ではこれらの話題について説明する。

前章で述べたように携帯端末向けのテキスト表示では、ハイパーテキスト化とインタラクティブ性が重要である。その目的に沿ったPDA上のブラウザを紹介しておこう。

BuyukkoktenらのPower Browser³⁾は、PDAおよび携帯電話でWebページを表示するブラウザであり、要約とキーワードと漸進的表示(progressive display)を特徴とする。まず、WebページからHTMLのタグを見て均質な部分(段落、Table、リストなど)を認識する。これはSemantic Textual Unit (STU)と呼び、段落、リスト、など意味的にまとまりのあるテキストの部分である。ブラウザではSTU単位で表示をする。PDAに表示した例を図-2に示す。

図-2で左端が○の行は、その行に対応するSTUが全部表示されている。●の行は、STUの最初の1行のみが示され、実際のSTUはもっと長いことを示す。Power Browserではこの1行表示を1行要約と呼ぶ。1行要約の行の●をクリックすると、次は3行要約、すなわち最初の3行が表示される。もし、STUが3行以内なら○になるが、3行以上だと●が半分だけ白くなる。これをもう1回クリックするとSTU全文が表示され、○になる。この様子を図-3に示す。このほかにユーザの入力したキーワードを含む文の要約を表示することもできる。このブラウザは要約としてはSTUの先頭部分を表示するといういたってシンプルなものだが、それでも57%のブラウズ速度向上、75%の入力手間の軽減を実現している。

この章で述べた可読性の実験とPower Browserは、自然言語処理とは関連が薄いですが、携帯端末へのテキスト表示を検討する際には有益な知見を与えてくれる。



図-2 Power Browser の画面例

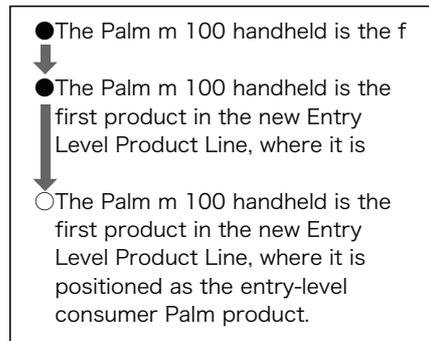
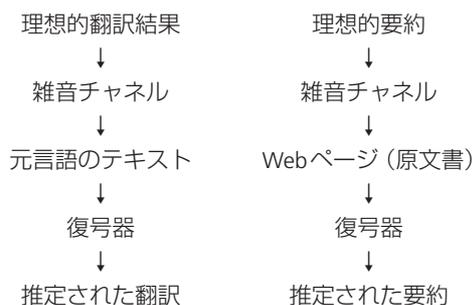


図-3 要約の状態変化

さて、このモデルを改善し、Webページの要約を生成するOCELOTというシステムの研究がなされた²⁾。対応コーパスにはWebページとその人手要約であるOpen Directoryを用いている。OCELOTの改善点は、要約で原文書(=Webページ)中出现した単語を直接使用のではなく、対応コーパスにおいて、要約と原文書で共起確率の高い単語を用いることである。これは統計的機械翻訳モデルとのアナロジーでいえば、



ということになり、同じ計算モデルや推定アルゴリズムが使える。さらに要約と原文書の対応付けも考慮することになる。そこで、OCELOTのモデルでは、ベイズの定理と機械学習の1つであるEMアルゴリズムを用いて元文書から要約を推定する。評価としては、このモデルによって生成されたK語要約が対応コーパスの要約に出現する単語をカバーする割合は、次の通りである。

K	4	5	6	7	8	9
カバー率	.41	.35	.30	.27	.25	.23

OCELOTによって生成された要約例を示す。

[Open Directoryの要約]

To advocate the rights of independent music artists and raise public awareness of artists distributing their music directly to the public via the internet

[OCELOTによる要約]

The music business and industry artists raise awareness rock and jazz

要約

携帯端末向けの表示では、1画面で内容が把握できることが理想である。従来のテキスト要約は文抽出を主体とし、場合によっては読みやすいテキストへの編集プロセスを組み合わせるものであった。これはinformativeな要約である。携帯端末表示向けの場合、抽出した文をさらに削り込むような処理が必要になる。一方、元のテキストへの内容示唆のみを与えるindicativeな要約はかなり短いテキストにできる。したがって、indicativeな要約の一種であるタイトルやheadline生成のようなタイプの要約が有望である。

従来のテキスト自動要約の研究では、携帯端末表示向けの要約を直接に標榜する研究は少ない。そこでこの節では、まず生成型の要約技術、次に抽出、編集方法の要約技術のうち、Webページを対象にした研究について紹介する。

◎生成型要約技術

原文書と、それに対応するheadlineのペアからなる対応付けコーパスを用い、本文とheadlineに共に現れる単語の対応の確率モデルを学習し、その結果に基づいて、headlineを自動生成する方法¹⁾が提案されている。この研究は(1) headline生成のために本文中に出現した単語だけを用いて要約を生成する、(2) 統計モデルとしてN-グラムを用いて生成された文の良さを評価する、モデルである。このモデルに従って生成したheadlineが含む単語が対応付けコーパスにおけるheadlineの単語をカバーする確率は、1語のheadlineで0.374、2語で0.248、以下徐々に下がり、6語で0.208程度となりあまり芳しくない。そこで、単語の出現位置情報と、品詞情報も加味したモデルにしたところ、0.02から0.04程度の向上が見られた。

文としての質にはまだ問題があるが、短くかつ indicative であるという点では、携帯端末表示向けの要約に近づいている。

◎重要個所抽出型

ここでは、原文書から抽出した重要文からさらに重要個所を抽出(裏を返せば不要個所の削除)を行い、携帯端末表示に適するほどに短文化する技術を紹介する。

望月らは、体言と用言の組合せ(名詞句や動詞句)を基本単位とする重要個所抽出方法を提案している⁶⁾。文章中の各基本単位に、(1)その基本単位に連なる基本単位でも同じ語が連続して現れる度合いと(2)構文的な重要度と加えた重み付けをし、重みの大きいものを選択する。選択された基本単位の意味を成り立たせるために必要な他の基本単位を、構文解析結果の係り受けを利用して認識し追加する。こうして選んだ重要個所に含まれる語の語彙連鎖重要度を減少させ、再び同じ方法で基本単位を選択する。これを定められた要約率になるまで繰り返す。

システムの要約と人間の要約との単語頻度ベクトルの類似度で評価した結果、要約率 0.3 程度で類似度は 0.5 程度である。

中川らは非重要個所削除する方法を提案している⁷⁾。コーパスとしては毎日新聞社からインターネットに配信されている記事集合を用いる。

まず同じ日に発信されたインターネット記事を文書集合と見なし、名詞の $tf \times idf$ を計算する。なお、 $tf \times idf$ とは、ある単語の文書における出現数に比例し、少ない文書にだけ現れると大きくなるという性質を持つ重み付け関数である。要約対象のインターネット記事の第 1 段落の文を係り受け解析する(なお、以下の実験結果は、京大で開発された KNP による)。

係り受け解析の結果の枝において、枝端にある名詞の $tf \times idf$ が小さい枝から刈りとり、所定の長さに要約する。所定の長さとしては、携帯端末表示を想定し、100 文字とした。

要約の一例を示す。

[例：元のインターネット記事の一部]

ヘリコプターと小型機が衝突、山中に墜落し、4人が死亡した事故はその後、2人の死亡が確認され、死者が6人となった。

係り受け解析の結果は図-4 のようになる。

下線を付けた部分が見出しおよび $tf \times idf$ の高い名詞である。名詞に下線が付いていない枝は重要ではないと見なして削除した結果、以下のように要約を生成する。

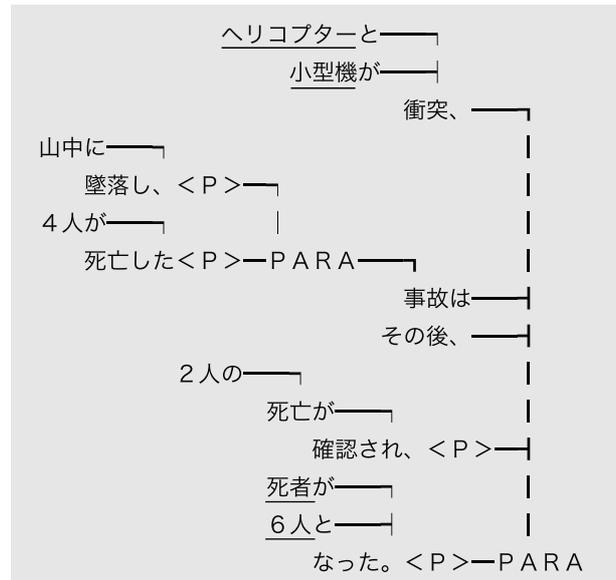


図-4 係り受け解析結果

ヘリコプターと小型機が衝突、死者が6人となった。

この方法の特徴は、原文書と要約の対応コーパスとして、インターネットに毎日配信されている毎日新聞の記事と、同じく毎日新聞から配信されているi-モード向け記事を用いて評価することにある。その結果、小規模な実験で、i-モードの記事に出現する単語を、提案方法の50文字以内の要約において60%程度カバーする。

抽出された要約文の編集の研究も盛んになってきている。そのうち、要約文から不要個所を削除する研究としては⁵⁾がある。抽出された要約文から、以下の不要部分を削除する。すなわち、(1)構文解析木を調べて、文法的に必須の部分(文の主動詞、名詞句の主辞など)ではない部分であり、かつ(2)主題に関係の深くない部分を削除する。主題との関係の深さは同じ語の繰り返しや同義語の使用頻度が高い語を関係が深いと見なす。(3)さらに人間が要約した場合に削除された句も削除する。ただし、これは文脈における削除の確率によって決める。たとえば、主動詞がgiveの場合はwhen節が削除される確率が高いなら、同じ条件でwhen節を削除するなど。この方法によって人間が42%削除するところを、この機械的方法でも32%は削除できた。

以上紹介した研究は、中川らの研究を除けば、携帯端末表示を直接意識した要約方法というわけではないが、要約文の長さをかなり自由に短くできる技術であり、携帯端末表示向きテキスト生成に応用できるものであろう。

縮約

従来の自動要約とは異なるが、言い換えによってテキストを短くする技術も携帯端末表示向きテキストの生成では重要な技術である。まず、縮約 (compaction) という技術を紹介し、次により広く「言い換え」という文脈で論ずる。

◎縮約 (compaction)

Corston-Oliverはマイクロソフトの携帯端末向け電子メールテキスト縮約ソフトを提案している⁴⁾。このシステムでは構文解析した結果の葉ノードを縮約する。縮約とは、たとえば、November→Nov., IBM Ltd.→IBM, Monday 15 January 2001→1/15/2001のような言い換えで表現を短縮することである。さらに、英語において単語内部の母音の省略も縮約の一種でたとえば、example→exmpleのように言い換えられる。また、冠詞の省略も行われる。このような処理によって、30%から50%、平均で40%ほど文長が圧縮される。しかし、圧縮の結果としてPrblmOfAutmtcSmmrztznのような文字列が得られるが、まるで暗号のようである。正しく解読できるのは50%という状況である。ちなみに、上記の文字列は "Problem of automatic summarization" である。

◎言い換え

上記の例は一種の言い換えによる圧縮である。日本語でも種々の言い換え、すなわち同じ意味を伝える別の言語表現に換えることが可能であり、最近、自然言語処理の1分野として研究が盛んになってきた。すでに述べたように、時刻、日付などは行末で分断されると読みにくいので、縮約の項で述べたような言い換えで分断を避けることが望ましい。一方、地名、人名、組織名も短縮形に言い換えることは携帯端末表示では有力と考えられる。しかし、短縮形への言い換え可能性は文脈依存的である。たとえば、「小泉首相」を「首相」と言い換えられるのは、首相が他にいない場合であって、サミットの記事で各国首相の名前が出る場合は「首相」と言い換えることはできない。

以上は名詞の言い換えだったが、文末表現の言い換えも有力である。すなわち、「国会で審議」のような体言止め、「民営化へ」というような助詞止め、「民営化？」のような記号止めがあり、いずれも文末表現が短縮される。元来の表現は「国会で審議に入った」「民営化へ向けて進む」「民営化されるかもしれない」のようになり長いものであろうから、文の短縮効果が大きい。また、個人的な感覚でも少ない文字数かつリズムミカルで直感的に理解しやすいように思う。実際に我々がi-モード記事を調べたところでもこれらの文末表現は多用されて

いる。2001年5月10～11日の毎日新聞のi-モード記事(経済、国際、政治、社会)においては、体言止め43記事、助詞止め16記事、文末が用言21記事であった。このことから見ても体言止め、助詞止めの行われる場合、可能な場合についての仕組みを明らかにすることは重要なテーマである。

展望

◎現在まで技術の成果のまとめ

1行あたり10から20文字の携帯端末画面への表示を念頭においた文書要約技術は、必ずしも明確な目標として意識された研究が多いわけではなかった。Power Browserのようなインタフェース重視型のシステムは確かに一定の成果が期待できるが、自然言語処理をしていないので、改善の余地はある。一方、OCELOTのような統計処理によるheadline生成は、いまだ十分な質の結果を達成していない。現在のところでは、やや古典的ではあるが、重要個所抽出型の要約が有望と思われるが、このようなシステムによって自動要約された結果を、人手で書かれた携帯端末向けコンテンツと比較することによって定量的評価を行うことが重要な時期にきていると考えられる。

◎要約配信システムのアーキテクチャ

携帯端末表示向けの要約を考えると、忘れてならないのがシステムの全体構成である。現在の携帯端末の能力を考えれば、携帯端末上で要約などの重い処理をするのはまだ無理である。したがって、サーバ側ないしはサーバとネットワークの間にproxyをおいて処理するのが現実的である。要約処理を行うタイミングも重要である。オンデマンドで要約をする処理は処理件数が増えると、まだまだ計算機の負荷が大きい。オフラインで要約を自動的に作っておく方法も有力である。

参考文献

- 1) Banko, M., Mittal, V. and Witbrock, M.: *Headline Generation Based on Statistical Translation*, 38th ACL, pp.318-325 (2000).
- 2) Berger, A.L. and Mittal, V.O.: *OCELOT: A System for Summarizing Web Pages*, 23rd ACM SIGIR, pp.144-151 (2000).
- 3) Buyukkokten, O., Garcia-Molina, H. and Paepcke, A.: *Accordion Summarization for End-Game Browsing on PDAs and Cellular Phones*, ACM SIGCHI'01, pp.213-220 (2001).
- 4) Corston-Oliver, S.: *Text Compaction for Displaying on Very Small Screens*, In *Proceedings of the Workshop on Automatic Summarization*, NAACL (2001).
- 5) Jing, H.: *Sentence Reduction for Automatic Text Summarization*, In *Proceedings of ANLP 2000*, pp.310-315 (2000).
- 6) 望月 源, 奥村 学: *読みやすさの向上と冗長性の排除を考慮した重要個所抽出型要約*, 情報処理学会NL研, 139-3, pp.17-24 (2000).
- 7) 中川裕志: *携帯端末向けコンテンツ記述*, 言語処理学会第8回年次大会ワークショップ「社会情報基盤のための言語・メディア処理」論文集, pp.33-40 (2002).

(平成14年10月23日受付)

