機能語の補完を用いた濃縮還元型要約モデル

池田 諭史 , 牧野 恵 , 山本 和英
長岡技術科学大学 電気系
E-mail:{ikeda,makino,ykaz}@nlp.nagaokaut.ac.jp

1 はじめに

現在、様々な要約の手法が研究されている。例えば、堀ら [4] は、単語重要度を最大にし、かつ日本語として自然な部分単語列の抽出を動的計画法によって行っている。田中ら [6] は、文書要約において重要語を決定し、その単語の必須格等を要約要素語として抽出して文としている。これはその文について不要部を削除することで要約を行なっている。これらは原文に存在する単語のみを用いて要約を行っている。しかし人間が要約を行う際には、自立語の換言のみではなく、機能語の変更も行い可読性の高い要約を行っている。例 1 は実際に Web のニュース記事と人手で要約した文の一部である。この例から人間は、自立語の換言のように機能語についても言い換える場合があることが分かる。

例 1) … ダイエー について、支援 を 決定した場合でも、…
→ … ダイエー への 支援 が 決定した場合でも …

さらに、この例で言い換えられている「について → への」は常に同じ意味で使われるわけではなく、前後の文脈から人間には同じ意味で使われていると判断が可能であり、常に言い換えできるわけではない。

そこで、人間が要約を行う際にどのような手順で行っているかを考える。人間が要約する方法はいくつか考えられる。その一つとして、要約に必要な単語をいくつか抜き出し、その単語群を用いて文を生成することによって要約を行なうことがある。我々は、自動要約でもこの方法を採用することが可能であると考え、原文から単語を抜き出し (濃縮)、その単語群より文を生成する (還元) ことで要約を行う手法を提案する。

我々[5] は以前にこのモデルによる要約手法を提案した。本手法との違いを 3 節で述べる。

2 提案手法

本稿で提案する手法は、以下の５つの処理で要約を行う。それぞれの処理部について以下の節で述べる。ここでの処理は上に示したものから順に行なう。

1. 前処理部 (2.1 節)
2. 単語抽出部 (濃縮部)(2.2 節)
3. 複合語の同定 (2.3 節)
4. 文生成部 (還元部)(2.4 節)
5. 生成文の再スコア付け (2.5 節)

2.1 前処理部

前処理部では、不要部分の削除と文末の整形を行う。ここでの不要部分は括弧部分である。文末の整形は述語の抽出のために行う。述語は文中で最も内容を表す語である。そこで単語抽出部 (2.2 節) で必ず抽出する。

2.1.1 不要部分の削除

本節では括弧部分の削除を行なう。船坂ら [2] によると新聞で使われる括弧の用法は説明用法と言い換え用法である。説明用法は「同僚の男（４５）」の様に括弧前の単語に説明を加える。この括弧部分は要約としては削除すべきである。言い換え用法は「日本農林規格（ＪＡＳ）」の様に括弧前の単語を括弧内の単語で言い換える。ここではどちらかの単語のみを使用すれば良い。また括弧内の語の方が短いことが多い。要約としては括弧内の語を残して括弧の前の単語を削除すべきである。しかし、括弧の用法の同定と言い換え対象部分の同定が必要となる。本節の処理は前処理として行っているために、ここでの誤りは後の処理に大きく影響する。そこで全ての括弧を削除することとした。

2.1.2 文末の整形

文末の整形は述語を同定する目的で行なう。ここで述語は一般的な述語ではなく、文の内容を最も表した単語を述語とする。例えば、例 2 での述語は「明らかにした」である。しかし、この文で重要な内容は「B さんが昇格した」ということであると考える。そこで本稿では例 2 の述語を「昇格」とする。本稿における述語は、文末整形後の文で一番最後の名詞、動詞とする。

例 2)A 社は B さんを 昇格 させたことを 明らかにした。

このような述語の同定は畑山ら [3] も行なっている。畑山らは本稿における述語を主動詞と呼び、パタンマッチによって同定を行なっている。我々はこのパタンを簡略化して用いた。我々が作成したパタンは 108 件である。本稿では、単語を抽出しその単語から文を生成することで要約を行う。原文には、要約に必要な情報が残っているならば、多少原文が日本語として間違った文になっても本モデルの要約には影響はないと考える。このパタンに当てはめた後に、我々[7] が提案した文末の整形手法を用いて文末の整形を行なった。この整形手法もパタンを用いて行なっている。

2.2 単語抽出部 (濃縮部)

ここでは、要約に必要な単語を抽出する。本稿で要約に必要な単語は名詞、動詞とした。2.1.2 節で同定した述語は必ず抽出する。単語抽出には 2 値分類の機械学習手法 Support Vector Machine(SVM) を用いた。SVM は教師あり学習なので学習データが必要になる。学習データには原文と要約文の対 (要約対) を用いる。原文より抽出単語候補である名詞、動詞を全て抜き出す。抽出単語候補が要約文に含まれているかを確認し、含まれていれば正例、含まれていなければ負例として学習データを作成する。素性は以下のものを用いる。カーネル関数は線形カーネルを用いた。

- 対象単語の表層形及び品詞
- 対象単語の前後 2 単語の表層形及び品詞
- 対象単語に直接係る単語の表層形及び品詞
- 対象単語が直接係る単語の表層形及び品詞
- 対象単語の前 1 単語が対象単語に直接係るか否か
- 対象単語の後 1 単語に対象単語が直接係るか否か

2.3 複合語の同定

本節では、単語抽出部で抽出した単語群から複合語として扱う単語について同定を行なう。ここで複合語候補となるのは、抽出単語群に含まれる単語を 2 個以上並べたもの全てである。2.3.1 節では複合語候補から複合語を決定するためのスコア (複合語スコア) を、2.3.2 節では単語抽出群に複合語を組み込む方法を述べる。

2.3.1 複合語スコア

複合語スコアは連接する２つの単語 A,B が複合語 “AB” であるかを判断するスコアである。複合語スコアは 2 単語 A,B が共起する文章における複合語 AB が含まれる割合である (式 (1))。このようなスコアを算出する際には、一般的に大規模なコーパスを用いる。しかし、複合語の選定をするのに十分な量のコーパスは手元には存在しない。そこで、検索エンジンを大規模なコーパスに見立てて、スコアの算出を行った。本稿で使用した検索エ

ンジンは Google[4] である。このスコアに閾値を設け閾値以上の複合語候補を複合語とする。これ以降、複合語とは複合語スコア (式 (1)) が閾値以上の語を指す。複合語の最後の単語を複合語の主辞と呼ぶ。複合語の品詞は主辞の品詞とする。

$$Score_{comp}(A,B) = \frac{|''AB''|}{|''A''\ and\ ''B''|} \quad (1)$$

$Score_{comp}(A,B)$: AB が複合語である複合語スコア

$|A|$: A の $Google$ でのヒット件数

2.3.2 単語群の複合語化

本節では、複合語を単語群に適用する。適用の際には、包含関係にある複合語は包含する側の複合語を採用する。例 3 では「公的年金」「年金制度」「公的年金制度」が複合語である。「公的年金制度」は「公的年金」「年金制度」共に包含しているので「公的年金制度」を複合語とする。

例 3){· · · , 公的, 年金, 制度,· · · }

次に、複合語化する際に起こる問題について説明する。例 4 では「米映画」「映画制作」「制作会社」の 3 つが複合語である。この 3 つの複合語はどれをこの抽出単語群で使用するかを決めることができない。例 4 で “-” で繋がれた複合語はどの複合語にするか決められない。これ以降 “-” で繋がれる複合語の数でこの問題の箇所を呼ぶ。例えば、例 4 は 3 語が “-” で繋がれているので、これを「3 つの複合語が連鎖して決められない場合」と呼ぶ。

例 4){· · · , 米, 映画, 制作, 会社,· · · }
→{· · · , 米映画-映画制作-制作会社,· · · }

このような複合語を決定できない箇所は 1 文に複数個存在することがある。この場合それぞれの箇所を独立に考える。

2.3.3 使用する複合語の決定

本節では、2.3.2 節で挙げた使用する複合語が決定できない際の決定方法について説明する。使用する複合語を決定する前に、2 つの複合語をまとめて 1 つにする場合がある。2 つ以上の単語が 2 つの複合語に共通して存在する場合は 2 つの複合語をまとめて 1 つの複合語とする。例 5 はその例の 1 つである。例 5 では「公的年金制度」と「年金制度改正」が複合語なので、本来はどちらかを複合語にする。しかし「年金制度」という 2 単語が共通しているのでこの 2 つをまとめて 1 つの複合語とする。この処理を最初に全ての複合語が決められない箇所について行なう。

例 5){· · · , 公的, 年金, 制度, 改正,· · · }
→{· · · , 公的 年金制度-年金制度 改正,· · · }
→{· · · , 公的年金制度改正,· · · }

次に、2 つの複合語が連鎖して決められない場合と、3 つ以上の複合語が連鎖して決められない場合の 2 つの場合に分けて処理を行なう。それぞれの場合の処理について述べる。

2 つの複合語が連鎖して決められない場合について述べる。ここで、前の複合語を複合語 A、後ろの複合語を複合語 B とする。以下の規則により複合語の決定を行なう。上に示した規則ほど優先度が高い。条件 3 の場合は文生成部 (2.4 節) で補完候補とする機能語をノ, ヲ, ニ, ガに限定する。全ての条件に当てはまらない場合は複合語スコア (式 (1)) の大きい方を複合語とする。

1. A の品詞が動詞である →2 つの複合語をまとめて 1 つにする
2. B の品詞が動詞である →A を複合語とする
3. B の品詞がサ変名詞、且つ A の品詞がサ変名詞以外
→A を複合語とする

3 つ以上の複合語が連鎖して決められない場合について述べる。ここではそれぞれの複合語の品詞がサ変名詞かそれ以外かに着目して処理を行なう。この処理で決定する複合語は 1 つである。以下の規則で使用する複合語を決定する。

1. 品詞がサ変名詞の複合語が 1 つ
→ 品詞がサ変名詞の複合語の前の複合語を使用する
最初の複合語がサ変名詞の場合最初の複合語を使用する
2. 品詞がサ変名詞以外の複合語が 1 つ
→ 品詞がサ変名詞以外の複合語を使用する
3. それ以外 → 複合語スコアの最も高い複合語を使用する

決定した複合語の前後の複合語では、決定した複合語と共通する単語が使用できなくなるので、複合語を決定できない箇所に変化が生じる。そこで、ここで使用することを決定した複合語は必ず、複合語として用いるという条件をつけて、2.3.2, 2.3.3 節の処理を行なう。この処理を複合語が決定できない箇所が無くなるまで行なう。

2.4 文生成部 (還元部)

ここでは 2.1〜2.3 節で作成された単語群から文を生成する。文の生成は機能語を補完することで行なう。本稿における機能語は助詞、助動詞とその連続とする。2.3 節で同定した複合語は主辞のみを用いる。また、本節ではコーパスから作成した様々な言語データを用いる。この言語データを作成する際にコーパス内の複合語 (名詞, 動詞の連続) は本節の複合語の扱いと同様に主辞のみを用いる。コーパス内の複合語は 2.3.1 節で同定した複合語ではなく名詞、動詞が連続したものを複合語とする。

2.4.1 補完箇所に対する補完候補の決定

本節では補完候補の出力方法について述べる。機能語を補完する箇所は抽出単語の間全てとする。最後の単語の後にも機能語を補完する。この機能語を補完する箇所を補完箇所と呼ぶ。また補完箇所に補完する機能語の候補を補完候補と呼ぶ。最初の単語の前には機能語を補完しない。これは、機能語から始まる文は日本語にはないためである。

補完候補の出力はコーパスを用いて行なう。補完箇所の前後の単語を用いてコーパスを検索し補完候補を出力する。例えば、補完箇所の前後の単語が「安全」と「検査」の場合には「安全+(機能語)+検査」となる機能語全てを出力する。ここでの機能語はコーパスに出現する助詞、助動詞の連続全てである。しかし助詞、助動詞の連続からなる機能語の中にはあまり使用しない機能語や、他の機能語でも言い換えられるものが存在する。そこでコーパス内の出現頻度に閾値を設けることにより補完候補とする機能語を限定する。2.3.3 節で補完候補を限定した場合はその限定を優先しノ, ヲ, ニ, ガのみ補完候補とする。

また、最後の単語については特殊な処理を行なう。これは最後の単語の後に補完される機能語は文末になるためである。最後の単語が名詞である場合は、その後ろに機能語を補完せずに体言止めの文にする。最後の単語が動詞でかつ文末に出現する動詞リストにある動詞であれば、後ろに機能語を補完しない。

ここで文末に出現する動詞リストについて述べる。動詞の中にも後ろに機能語を補完して文が終了する動詞と、補完せずに文が終了可能な動詞が存在する。本稿の目的は要約なので動詞の後に文を補完しなくても良い動詞については補完を行なわない。そこで機能語を補完しなくても良い動詞のリストを作成した。これが文末に出現する動詞リストである。コーパス内で文の最後に出現した動詞を全てを文末に出現する動詞とした。

2.4.2 機能語の補完を用いた文生成

補完候補からの補完語の決定を行なう。本稿では、補完語の決定をラベル付与問題にみたてて行なった。ラベル付与問題での観測を本手法の抽出単語群に、ラベル付与問題でのラベル列を本手法での機能語列とした。本手法の例を図 1 に示す。

この問題は式 (2) で求めることができる。また、確率値は信頼区間を用いて補正を行なっている。信頼区間の算出には Agresti and Coull[1] の手法を用いた。試行回数が少ないほど信頼区間が広くなる。信頼区間は 0〜1 の値が与えられるので、1 と信頼区

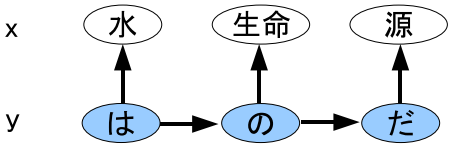

図 1: HMM を用いた機能語付与問題の例

間の差を不信頼度とする。ここ不信頼度と確率の積を取ったものを新たな確率とした[1]。

$$\mathbf{y} = \underset{\mathbf{y}\in\sum_y^T}{\operatorname{argmax}} \prod_{t=1}^{T} P(x_t|y_t)P(y_t|y_{t-1}) \quad (2)$$

T : ラベルの数

$P(x_t|y_t)$: 後方からの単語 $2gram$ 確率

$P(y_t|y_{t-1})$: 機能語の単語 $2gram$ 確率

生成文は A*アルゴリズムを用いて上位 N 件出力する。また実際に計算を行なう際は確率をコストに変換している。コストは確率の対数の絶対値である。このために実際には積の計算が和になり、最大値を求めるのではなく最小値を求める。

2.5 生成文の再スコア付け

本節では、2.4 節で求めた N 個の生成文に対して、新しくスコアを付与する。式 (2) で求めたスコアを連接スコアと呼ぶ。本節ではこの連接スコアの他に係り受けスコア、機能語スコアを用いて再度スコアを付与する。係り受けスコア、機能語スコアについて以下の節で説明する。

2.5.1 係り受けスコア

係り受けスコアの算出には構文解析器 CaboCha[2] を用いた。CaboCha の付与する係り関係スコアを用いる。この係り関係スコアは一般に係りやすさの度合いを示す。係りやすい文節が多い程、正しい文らしいのではないかと考えこの係り関係スコアを用いる。係り関係スコアは文節毎に付与される。係り受けスコアは、係り関係スコアを全て足し合わせたものとした。

2.5.2 機能語スコア

機能語スコアの算出には、述語に対する機能語の述語に対する機能語の使われ方を使用した。述語に対する機能語の使われ方はコーパスより算出する。単語の使われ方の算出方法を以下に示す。

1. コーパス全文に対し、文末整形 (2.1 節) を行なう
2. 整形後、構文解析し機能語と述語のみを抜き出す
3. 1 文毎に機能語の単語ベクトルを作成する
4. 同じ述語の単語ベクトルの算術平均をとる

これにより、述語に対する 1 文での機能語の使われ方が分かる。生成文に対し同様に機能語の単語ベクトルを作成し、コーパスで学習した同じ述語の単語ベクトルとのコサイン距離を求める。これを機能語スコアとする。

2.5.3 スコアの合成

3 種のスコアを用いて各生成文に対するスコアを計算する。3 種類のスコアはそれぞれの値が大きく異なるので大きさを合わせる必要がある。連接スコアと係り受けスコアは両方とも同じコストの形態である。機能語スコアだけ違うので機能語スコアを変換する。機能語スコアはコサイン距離なので 0〜1 の値である。これは確率と同じ扱いができる。そこで連接スコアと同様に対数の絶対値をとることでコストとする。これらのスコアにより、最終的なスコアを決定する。ここで全てのスコアは小さい方が良い。したがって、新しいスコアを付与し、最もスコアの小さいものを最終的な要約文とする。最終的なスコアは各スコアの重み付きの和である。重みは実験的に求める。

3 先行研究からの変更点

ここでは、以前に我々[5] が提案した手法 (先行研究) からの変更点について示す。

先行研究では 2.1,2.3,2.5 節を行なっていない。つまり 2.2,2.4 節のみで要約を行なっている。まず、2.2,2.4 節での処理の違いについて述べる。2.2 節での変更点は素性である。先行研究では係り受けを用いずに、tf と idf を用いていた。要約に必要な単語は使われ方が似ていると考えた。次に 2.4 節の変更点は 3 点ある。1 点目は複合語の同定をしていないので代りに、空文字を補完候補に導入したこと、2 点目は補完候補の出現頻度での足切りをしていないこと、3 点目は信頼区間による確率の補正をしていないことである。

4 評価実験

提案手法の妥当性を測るために、実際に実験を行なった。2.2 節では SVM で学習を行なっている。要約対は原文として NIKKEI NET[5] を、要約文として日経ニュースメール[6] を用いた。NIKKEI NET は日本経済新聞社が Web で配信しているニュース記事である。日経ニュースメールは Nikkei-goo の行なっているサービスである。これは新幹線の電光掲示板のニュースで用いられている記事をメールサービスとして配信しているものでありニュース文が非常に短い表現になっている。これら記事は同じ新聞社のニュース記事なので同じタイトルの記事が存在する。そこで、タイトルが一致した記事を記事単位の要約対とする。本稿では文要約を目的としているので文単位での要約対の方が望ましい。ニュース記事は重要な情報が最初にあるという特徴がある。そこで、これらのタイトルが同じ記事の 1 文目を文単位の要約対として利用する[2]。ここで集めた要約対は 3361 対である。これより無作為に 3300 対を取り出して用いた。文生成部 (2.4.2 節) での各種連接確率、及び補完候補の出力と生成文の再スコア付け (2.5.2 節) で用いたコーパスは日本経済新聞全記事データベース[7] を用いた。ここで用いた日本経済新聞は 1996〜2004 年の 9 年分である。

形態素解析には形態素解析器 ChaSen[1] を用いた。構文解析には構文解析器 CaboCha[2] を用いた。SVM 学習には TinySVM[3] を用いた。また CaboCha で固有表現タグが付いた単語はまとめて 1 単語とし、その固有表現タグに汎化している。本稿の機能語は助詞、助動詞が 1 個以上連続したものを機能語としている。

文生成部での出力は上位 100 件とする。

5 結果

実際に本手法で要約率が約 80%を目指して単語抽出率 70%の要約を行なった。要約した 100 文について、3 人の被験者が独立に評価を行なった。評価は生成された要約文が日本語として正しいか (評価 1) と、原文と比較して意味が保持されているか (評価 2) という 2 点について行なった。その結果を表 1 に示す。

表 1: 正解の人数を変えたときの正解率

正解とした評価者数	≥ 1	≥ 2	$= 3$
可読性の評価 (評価 1)	66%	38%	20%
意味同一性の評価 (評価 2)	56%	23%	3%

人による揺れが大きいことが分かる。また意味同一性の評価では、正解とした人数を変更することで非常に大きな揺れが見られる。単語抽出部 (2.2 節) で抽出した単語群を用いて人間が文生成した場合でも意味同一性の評価は約 40%である。

[1] 総和が 1 になる保証がないので厳密には確率ではない。

[2] これ以降文単位の要約対を要約対と表記する。

6 考察

6.1 単語抽出部の精度

単語の抽出数を変更したときの単語抽出の精度を求めると図 2 となった。ここで示す結果は要約対 3300 対を用いた 5 分割交差検定によって得られた精度の平均値である。

これより抽出精度は原文の名詞、動詞の数の 80%を抽出したときがもっとも良い精度になっていることが分かる。また、図 2 にはベースラインとして、先行研究 [5] での素性の抽出と $tf \cdot idf$ を用いて重要度が高い単語から抽出した場合の F 値も描いている。この結果を見ると本手法は $tf \cdot idf$ より高い精度で単語の抽出が行われていることが分かる。また先行研究の手法とはほとんど精度に差がない。また同じデータを使ってるにも関わらず、100%抽出したときの値が違うのは、2.1 節で括弧や文末を削除した後の文から割合を求めているためである。

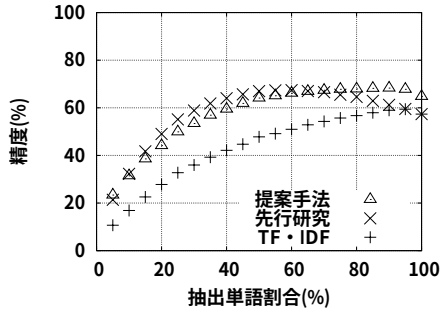

図 2: 提案手法での単語抽出割合における抽出精度の変移

またこの結果からは先行研究との差は分からない。そこで、実際に先行研究での抽出単語群と本手法での抽出単語群から人手で文生成を行ない、比較した。これは 2 人の被験者が文を生成し、他の 1 名の被験者が評価をした。用いたのは上記で評価した 100 文である。2 人の被験者の平均を示す。人間が生成したのに可読性の評価 (評価 1) が 100 文でないのは、文生成できない単語群が存在したためである。

表 2: 2 人の被験者による文生成の結果 (平均)

手法	提案手法	先行研究
可読性の評価 (評価 1)	74 文	62 文
意味同一性の評価 (評価 2)	44.5 文	39 文

この結果より、先行研究時に比べてかなり多くの文が生成可能になっている。これには 2 つの要因が考えられる。1 つは変更した素性の問題である。変更した素性は tf と idf の情報ではなく、係り元、係り先の情報を用いている。これは、文要約を作成する際には、重要な単語よりもがその単語の使われ方の方が重要になると考える。もう 1 つは述語の問題である。これは、述語が得られたことで日本語らしい文が作りやすくなったと考える。

6.2 再スコア付けの効果

2.5 節で新しくスコアを付けた際の効果について調べる。再スコア付けして並べ替える前と並べ替えた後の精度を調べる。ここで、文生成時の出力は上位 100 文とし、他の評価と同じ 100 文を用いた。正解が上位 100 文内に存在する文のみを用いて評価を行なう。評価は正解が出現する順位 (評価 3) と正解が 1 位に出現する確率 (評価 4) とした。結果を表 3 に示す。この表は評価 1 で正解を判断した表である。

表 3: 並び替えの有効度

	評価 3	評価 4
並び替えせず	8.21	0.31
並び替えをする	6.57	0.44

この時、正解が上位 100 文内に含まれる文は 70 文だった。正解が最初に出現する順位が約 2 位、1 位に正解が出現する確率で約 10 ポイントの上昇が見られた。また、評価 2 で正解を判断した場合においても同様の傾向がみられた。評価 2 で正解を判断した際の上位 100 文内に正解が含まれる文は 41 文であった。

このことより、上位 100 文内に正解が存在すれば正解としたとき、その精度は人手で生成した場合 (表 2) とほぼ等しい。これは、文生成は最適なラベル列を求める問題としているために、上位 100 文を出力するということは足切りをしているということになる。人手で生成したときとほぼ同等数の正解を保持したまま足切りできているといえる。このことより、文生成の精度は高いといえる。

7 まとめ

原文から単語群を抜き出して、その単語群から文を生成するモデルを用いて要約を行なった。このモデルで要約を行なうために処理を 5 つに分けた。前処理、単語抽出、複合語の同定、文生成、再スコア付けの 5 つである。要約率 80%を目指して要約を行なった際の精度は可読性の評価 (評価 1) で 38%、意味同一性の評価 (評価 2) で 23%であった。

単語抽出の精度は先行研究よりも良くなっていることが分かる。また文生成部を最適なラベル列探索問題の足切りと考えると、生成の精度は高い。また、再スコア付けの効果もあることが分かった。

今回は大局的なスコアの導入を考えた。しかし、実際には以前に比べてスコアがより大局的になったが、完全に大局的なスコアとは言えない。そこで今後の課題として、さらに大局的なスコアの導入が挙げられる。また、今回も意味を考慮したスコアが導入できていないので、意味を考慮したスコアの導入が挙げられる。

謝辞

本研究の一部は, 科学研究費補助金 基盤 (A)「円滑な情報伝達を支援する言語規格と言語変換技術」課題番号 16200009 によって実施した.

使用したツール及び言語資源

1) 形態素解析器 ChaSen, Ver.2.3.3, 奈良先端科学技術大学院大学 松本研究室, http://chasen.naist.jp/hiki/ChaSen/,
2) 構文解析器 CaboCha, Ver.0.53, 奈良先端科学技術大学院大学 松本研究室, http://chasen.org/ taku/software/cabocha/
3) SVM 学習ツール TinySVM, Ver.0.0.9, 奈良先端科学技術大学院大学 松本研究室, http://chasen.org/˜taku/software/TinySVM/
4) 検索エンジン Google, http://www.google.co.jp/
5) NIKKEI NET, 日本経済新聞社, http://www.nikkei.co.jp/
6) 日経ニュースメール,NIKKEI-goo, http://nikkeimail.goo.ne.jp/
7) 日本経済新聞全記事データベース 1996–2004 年度版, 日本経済新聞社

参考文献

1] Alan Agresti and Brent A. Coull. Approximate is better than "exact" for interval estimation of binomial proportion. In *The American statistician : a publication for the American Statistical Association / American Statistical Association*, Vol. 52, pp. 119–126, 1998.

2] 船坂貴浩, 山本和英, 増山繁. 冗長度削減による関連新聞記事の要約. 情報処理学会研究報告 NL144-7, pp. 39–46, 1996.

3] 畑山満美子, 松尾義博, 白井諭. 重要語句抽出による新聞記事自動要約. 自然言語処理, Vol. 9, No. 4, pp. 55–73, 2002.

4] 堀智織, 古井貞熙. 係り受け SCFG に基づく音声自動要約法の改善. 信学技報, SP2000-116, pp. 245–250, 2000.

5] 池田諭史, 牧野恵, 山本和英. 濃縮還元型文要約モデルの検討. 情報処理学会研究報告 NL174-13, pp. 71–76, 2006.

6] 田中信彰, 面来道彦, 野口貴, 矢後友和, 韓東力, 原田実. 意味解析を踏まえた自動要約システム abisys. 自然言語処理, Vol. 12, No. 1, pp. 143–164, 2006.

7] 山本和英, 池田諭史, 大橋一輝. 「新幹線要約」のための文末の整形. 自然言語処理, Vol. 12, No. 6, pp. 85–112, 2005.