JAIST Repository
https://dspace.jaist.ac.jp/

Title	長さの異なる要約の自動生成
Author(s)	伊良波，隆
Citation	
Issue Date	1999-03
Type	Thesis or Dissertation
Text version	author
URL	http://hdl.handle.net/10119/1228
Rights	
Description	Supervisor:佐藤 理史，情報科学研究科，修士

JAIST
JAPAN
ADVANCED INSTITUTE OF
SCIENCE AND TECHNOLOGY

Japan Advanced Institute of Science and Technology

修 士 論 文

長さの異なる要約の自動生成

指導教官　佐藤理史 助教授

北陸先端科学技術大学院大学
情報科学研究科情報処理学専攻

伊良波 隆

1999 年 2 月 15 日

Copyright © 1999 by Takashi Iraha

要 旨

本研究では、ニュースグループ fj.sys.sun の要約を自動生成する。その際、長さの異なる２つの要約を生成する。

目 次

1 序論 **1**
1.1 研究の背景 1
1.2 研究の目的 1
1.3 本論文の構成 2

2 研究対象の調査 **3**
2.1 ニュースグループの質問記事の特徴 3
2.2 文のタイプと重要文 5
2.3 行為の７段階理論 7

3 重要文の抽出による要約生成システム **9**
3.1 要約生成システムの概要 9
3.2 文のタイプ判定 11
3.3 意味なしセンテンスの除去 13
3.3.1 意味なしセンテンスの定義 13
3.3.2 意味なしセンテンスの判定 13
3.3.3 意味なし名詞リストの作成 15
3.4 長さの異なる重要文の抽出 15
3.4.1 短めの要約の抽出 15
3.4.2 長めの要約の抽出 16
3.5 従来研究との比較 16

4 実験と検討 **18**
4.1 タイプ判定の評価 18
4.2 要約生成の評価 19
4.3 関連研究 20

5 結論 **21**

第 1 章

序論

1.1 研究の背景

現在、インターネットの普及により、ワールドワイドウェブ (World Wide Web)、電子ニュースなどの電子化されたメディアが増加している。電子化されたメディアはネットワークを通じて、地理的な制約や時間的な束縛を受けることなく様々な種類の情報にアクセスできる。そのため、非常に便利なネットワーク社会が構築されつつある。

しかし、利用できる情報が急速に増加することによって、ユーザは必要な情報を的確に捜し出すことが難しくなっている。そのため、有用だが利用されない「眠った」情報を増大させるという状況が引き起こされてきている。

求める情報を得るためにユーザは情報検索を行ない、どの情報が必要であるかどうかを決定しなければならない。その決定を支援する１つの方法として、簡潔で適切な要約を提示することが挙げられる [1]。そうすると、ユーザは全文を読まなくても済むため、必要とする情報を容易に取捨選択することができる。

本研究では、電子ニュースのニュースグループ fj.sys.sun を対象テキストとし、ニュース記事を自動的に要約するシステムを作成する。

1.2 研究の目的

佐藤研究室では、自動編集プロジェクトの一環として、質問応答パッケージ SUN QA-Pack[2] を開発し Web 上で公開している (http://www-sato.jaist.ac.jp:8000/faq/)。SUN QA-Pack とは、ニュースグループ fj.sys.sun （SUN ワークステーションに関する質問記事と、その応答記事とからなるニュースグループ）の質問記事を分類し、短いサマリーを

つけて求める記事を捜し出すことを支援するシステムである。

本研究では、長さの異なる要約を自動生成することを試みる。SUN QA-Pack では、それぞれの質問記事に対して１種類の長さの要約を生成していたが、これとは異なる長さの要約を生成することを実現する。

1.3 本論文の構成

本論文は以下のように構成される。まず、第１章では、研究の背景や目的について述べた。第２章では、本研究の対象であるニュースグループ fj.sys.sun の質問記事の調査を行ない、その特徴を明らかにする。そして、その特徴に基づいて、文のタイプの設定をする。第３章では、重要文の抽出による要約の生成方法について述べる。第４章では、実験とその評価を行なう。第５章で結論を述べる。

第 2 章

研究対象の調査

この章では、本研究の研究対象であるニュースグループ fj.sys.sun の質問記事の特徴について述べる。そして、文のタイプを設定し、どのような文が要約として抽出すべき重要文となるかを明らかにする。

2.1 ニュースグループの質問記事の特徴

ニュースグループ fj.sys.sun は、質問応答型のニュースグループで、主に、SUN ワークステーション関連の、ハードウェアやソフトウェアに関する質問記事とそれに対する応答記事から構成される。ここでは、1997 年 1 月の質問記事 91 記事を調査した。その結果、質問記事は次の 2 つのタイプに大きく分類できることがわかった。

- **失敗しました型記事**
  失敗しました型記事とは、「インストールできない」、「エラーがでる」、「うまく動作しない」等、なんらかのトラブルが発生し、その解決方法を質問する記事のことである。例を図 2.1 に示す。

- **探しています型記事**
  探しています型記事とは、「こういうソフトウェアはありますか」、「○○についての本はありますか」といった、ソフトウェアや本などの所在を尋ねる記事である。例を図 2.2 に示す。

行	センテンス
1	はじめまして、山田@阪大です。
2	初めて投稿します。
3	UNIX などほとんど触ったことはなかったのですが、研究室の SUN の ss20 に Samba をインストールしろといわれてやってみたのですが、エラーメッセージが出てコンパイルの途中で止まってしまいます。
4	SUN OS のバージョンは 5.4 で Open Windows のバージョンは 3.4 です。
5	コンパイラーが悪いからだめなんでしょうか？
6	コンパイラーを gcc とかに変えなければいけないんでしょうか？
7	それとも、Makefile の設定を間違っているのでしょうか？
8	何も分かっていない素人ですが、よろしくご教授下さい。
9	お願いします。

図 2.1: 失敗しました型記事

行	センテンス
1	はじめまして。
2	Solaris2.x 用の日本語 ScalableFont を探しているのですがどなたかご存じないでしょうか？
3	最初から付いているものは２種類しかないですよね？XLoadFont で使用できる Font でかなり太めの物を探しています。
4	または Windows などの Font を f3b の形式に変換する TOOL などはないのでしょうか？
5	ご存知の方があれば教えてください。
6	よろしくお願いします。

図 2.2: 探しています型記事

2.2 文のタイプと重要文

fj.sys.sun の質問記事には典型的な文表現が数多く見られる。現れる典型的な表現は、上記の２つのタイプでそれほど違いはない。本研究ではこれらを１０種類に分け、以下のような文タイプとして整理した。

1. **挨拶、自己紹介**
   記事の先頭に書かれることが多い。
   (例) ・ はじめまして、小島@阪大です。
   ・ 谷川と申します。

2. **環境**
   投稿者が用いているマシン名、OS、ソフトウェアなどの情報を表す文。「〜を使用しています」、「環境は〜」という表現がよく用いられる。
   (例) ・ SPARC station-5 で Solaris2.5.1 を使用しています。
   ・ 環境は、Hayes ESP + Solaris2.5 x86 です。

3. **Goal**
   投稿者の目標（何をしたいのか）を表す文。意志を表す助動詞「〜たい」がよく用いられる。
   (例) ・ Solaris2.5.1 に gcc のインストールを試みています。
   ・ ファンクションキーにコマンドを登録したい。

4. **Fail**
   投稿者が直面した失敗、トラブル、エラーなどを表す文。否定文であるか、あるいは、文末に否定的な表現を表す動詞（ex. 「困っています」）が使われることが多い。
   (例) ・ 現在、C++ でコードをかいているのですが、SUN の ProWorks C++ の new の動作で困っております。
   ・ HD からブートすると、elx0: plumb: no such interface と出て、NIC を認識してくれません。
   ・ 早速ですが今 SS20 コンパチ機+Solaris2.5+X11R6.1 に Canna32p2 をうまくインストールできずに困っています。

5. **Question**
   直面した問題に対して回答を求める文。ほとんどの場合、疑問文となる。

(例) ・ どなたか、製品コンパイラなしで、R6.3 のコンパイルに成功された方はいらっしゃいますか？
・ POP3 用の server で、APOP をサポートサポートしているものはないでしょうか。

6. **Error または program の表示**
表示されたエラーや動かないプログラムを、参考のために載せた部分。英数字だけで記述される場合がほとんどである。
(例) ・ make: *** [all] Error 1 Could not build INN. Fix any problems and start again

7. **分析**
失敗した原因を投稿者が分析している文。 推測を表す「〜ようです」のような文末表現が使われるか、あるいは、「たぶん」のような副詞が使われることが多い。
(例) ・ エラーの原因は、ライブラリー作成時に、exportlistgen なるものをよびだすのですが、この中で c++filt というものを呼び出していて、これは、Sparc C Compiler 附属のもののようです。
・ たぶん HDD を見に行っている時に止まってしまっています。

8. **終りの挨拶**
文章の終り付近に書かれる締めの挨拶や言葉。
(例) ・ 情報不足があるかもしれませんが、よろしくお願いいたします。
・ 浅い内容で恐縮ですが、どうか宜しくお願い致します。

9. **Signature**
署名、投稿者の名前、メールアドレス。

10. **その他**
上記 1〜9 に当てはまらない文。

fj.sys.sun の質問記事において中心となる情報は、「何を行ないたいのか」、「直面した問題は何か」、「質問は何か」ということである。これらの文のタイプは **Goal** 、**Fail** 、**Question** である。従って要約を生成するには、これら 3 つのタイプの文を抽出すれば良い。

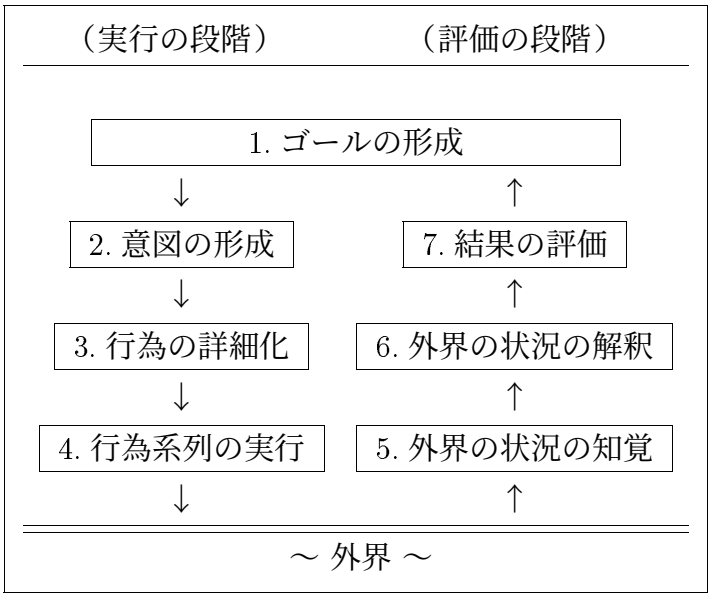

図 2.3: 行為の７段階理論　　D.A. ノーマン「誰のためのデザイン？」より

2.3　行為の７段階理論

この節では、fj.sys.sun の質問記事に見られる文タイプと、ノーマンの「行為の７段階理論」[3] を比較し、それらの対応を考察する。

ノーマンはユーザーインターフェースのよりよいデザインのために、人はどのように作業をするのかという観点で、「行為の７段階理論」と呼ばれる近似的モデルを作り考察している。行為の７段階理論の概要を図 2.3に示す。

1. ゴールの形成 : 起こって欲しいこと。

2. 意図の形成 : ゴールを達成するための意図を形成する。

3. 行為の詳細化 : 実行しようと計画している実際の行為系列。

4. 行為系列の実行 : 行為系列を実際に実行し、外界に働きかける。

5. 外界の状況の知覚 : 実行した結果、外界の状態を知覚する。

6. 外界の状況の解釈 : 予期に基づいて知覚を解釈する。

7. 結果の評価 : 起こると思っていたことに照らして解釈を評価する。

厳密な心理学的理論ではないので、７段階の区分は明確に区別する必要はなく、どこから始まってもよいし、全てを経由する必要もないとノーマンは定義している。

前節では、文のタイプを１０種類考えた。ここでそれらのタイプと７段階理論の対応を考えてみよう。これらの間には、以下のような対応関係が見られる。

- [1. ゴールの形成] と [2. 意図の形成] を明示的に記述した文が、**Goal** 文である。
- [5. 外界の状況の知覚] の状況を記述したのが、**Error** 文である。
- [6. 外界の状況の解釈] の状況を記述したのが、**分析**文である。
- [7. 結果の評価] において、それが否定的評価であった場合、それを記述すると **Fail** 文となる。
- 実行の各段階で次の段階へ進めない場合に、その事実を記述すると **Fail** 文（「〜できない」）となり、それが可能であるかどうかを質問するという形となると **Question** 文（「〜可能でしょうか」）となる。

要約は、そのテキストが表す情報の、(1) 概略情報、あるいは、(2) 中心情報、のいずれかを含むことが求められる。行為の７段階理論に基づくと、(1) の概略情報に対応するのは、行為の出発点となる [1. ゴールの形成]（**Goal** 文）であり、(2) の中心情報に対応するのは、行為の不成功を表す [7. 結果の評価]（**Fail** 文）、あるいは、次の段階を進めない事実を表す **Fail** 文と **Question** 文である。このことからも、文のタイプが **Goal**、**Fail**、**Question** の文を抽出するのがよいことが示唆される。

第 3 章

重要文の抽出による要約生成システム

本章では、２章で設定した文のタイプをもとに重要文を抽出する、要約生成システムについて述べる。

3.1 要約生成システムの概要

要約生成システムの処理の概要を図 3.1 に示す。

1. **記事本文の入力**
   記事を１文毎に分割し読み込む。

2. **文のタイプ判定**
   ２章で設定した文のタイプをもとに、読み込まれたそれぞれの文がどのタイプであるか判定する。

3. **意味なしセンテンスの判定**
   情報を含まない文を判定し、それらを抽出しないようにマークアップする。

4. **重要文抽出**
   ここでは、長さの異なる以下の２種類の要約を抽出する。

   - **短めの要約**：１文を抽出
   - **長めの要約**：数文を抽出

以下では、文タイプの判定、意味なしセンテンスの判定、重要文抽出の詳細について述べる。

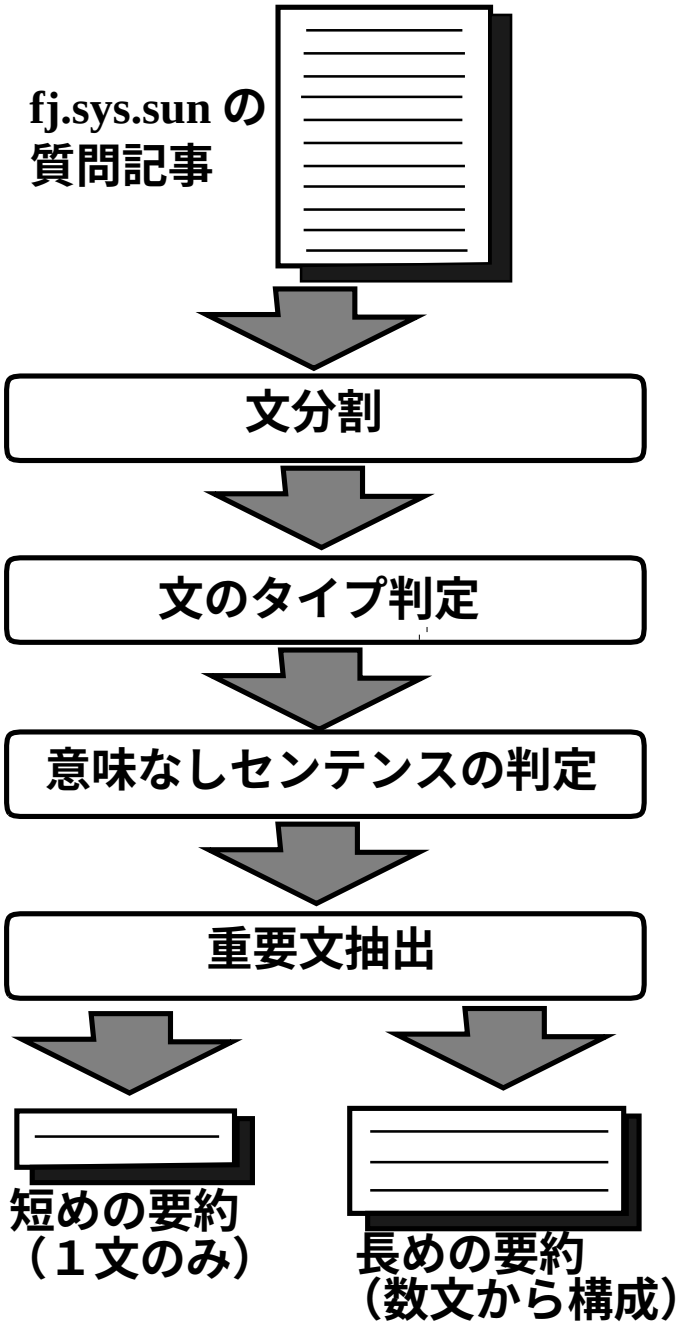

図 3.1: 要約生成システムの処理の概要

3.2 文のタイプ判定

２章で設定した文のタイプをもとに、読み込まれた１文がどのタイプであるか判定する。判定には、各々の文タイプに対して用意した文字列パターンを用いる。使用した文字列パターン（Perl の正規表現）の一部を以下に示す。

1. **挨拶、自己紹介**

   m/(いいます|(申|もう) します|はじめまして|(@|©)|投稿|
   ((宜|よろ) しく|)(お|御)(願|ねが) い (します|致します|いたします)|
   (御無沙汰|ごぶさた) して (おり|)(ます|)|クロスポスト|
   お世話になって (おり|)(ます|)|(こんにち|今日)(は|わ)|
   (今晩|こんばん)(は|わ))/

2. **環境**

   m/(Solaris|gcc|SS20|HP|Openview|Sparc|SUN|SunOS|SUNO535|メモリ (ー)?|
   sendmail|netscape|linux|LasarWind|DiskSuite|BSD|環境
   を (使|つか)ってい (ま (す|した)|る)|を (管理|かんり) してい (ます|した|る)/

3. **Goal**

   m/(行|おこな) おうと (思|おも|)(う|います|)|(行|おこな) いたい|しようと|
   (気持|きも) ちにな|する予定 (です|だ)|(検討|けんとう) して (いる|います)|
   (考|かんが) えて (いる|います|ます)|(させ|し|やり|やってみ|
   ((知|し) り|(調|しら) べ)|(手|て) に (入|い) れ|使い) た (い|くて)|
   (試|こころ) み (て|たい|よう|た)|(した|やっ(た|てみた))(ところ|のですが))/

4. **Fail**

   m/(((出来|でき)(な|ま)(い|く|せん))(?!か)|
   ((分|わ) か|判|解)(りません|らな (い|かった))|
   うまく (い|行)(きません|かない)|(表示|copy|コピー)?されません|
   しろと (言|い) われます|してくれません|しない状態になって|
   (止ま|立ち上が) らな|なってしま (う|った|いました)|
   できない|されてしま (う|った|いま (す|した))|開かない|

おかし (い|く)|使え (ません|なく)|問題が生じて|(困|こま)っ(た|て)|
(動|うご) きません|(機能|動作|起動) し (て|)(おりません|いない|ない|ません)|
(悩|なや) んでいます|を (起|お) こす|(落|お) ち (てしまい|) ます|
(し|) てしま (っ(た|て)|いま (した|す)|う)|してもダメ|つまづいています|
(エラー|error|(エラー)?メッセージ)(が|に)(で|出|な|なり)(る|た|て|ます|ました|る|った)|遅くなっている|エラー)/

5. **Question**

   m/(でしょうか|ですか|ますか)(?!(ら|ね))|いらしゃいま (すか|せんか)|
   (教|おし) え (て|)(ください|ほしい|いただければ|)|ですよね|
   探して (い|おり) ます|ありますか|ご教示下さい|
   あるのか|((御|ご) 教授|お知らせ)(いただければ|(下|くだ) さい)|
   お聞かせ下さい|お聞きしたい)/

6. **Error または program の表示**
   英数字だけからなる文をこのタイプと判定する。

7. **分析**

   m/(の (よう|様)(です|だと)|(たぶん|多分)|と思 (う|い|われ)(ます|ました)|
   思え (ます|る)|だろう|(考|かんが) えられます|(おそ|恐) らく)/

8. **終りの挨拶**

   m/((終り|おわり)|かしこ|情報を (お|御)(願|ねが) いします)/

9. **Signature**

   m/(- --|--)/

10. **その他**

もし、ある文が上記の 1 から 9 にパターンのうち、複数のパターンとマッチした場合は、その文は、複数のタイプを持つ文となる。

3.3 意味なしセンテンスの除去

前節の文タイプ判定では、意味のある情報を全く含まない文が **Goal**、**Fail**、**Question** に判定される場合がある。以下では、このような文を意味なしセンテンスと呼ぶ。本節では意味なしセンテンスを除去する方法を述べる。まず、意味なしセンテンスとは何か定義する。次に、意味なしセンテンスを判定する方法について述べ、最後に、この方法で利用する「情報を含まない名詞リスト」を作成する方法について述べる。

3.3.1 意味なしセンテンスの定義

「意味なしセンテンス」とは、意味のある情報を含んでおらず、重要文として抽出するのは不適切である文のことをいう。文のタイプが **Goal**、**Fail**、**Question** に判定された場合でも、意味のある情報を含んでいない場合がある。例を以下に示す。

- どこから手をつければいいのか分からない状態です。→ **Fail**
- ご存知の方がいらっしゃいましたら、お教え願えないでしょうか。→ **Question**

１つ目の例では、「何が」分からないのか、２つ目の例では、「何について」教えて欲しいのか、といった「何が」の部分が欠け落ちている。

何をしたいのか (**Goal**)、何で失敗したのか (**Fail**)、あるいは、何について聞きたいのか (**Question**) といった文は、fj.sys.sun における重要文である。しかし、その中に「何が」の部分がないものは、対象を特定する情報が含まれていないので、fj.sys.sun における重要文とは見做さないことにする。

3.3.2 意味なしセンテンスの判定

上に述べたとおり、意味なしセンテンスには、対象を特定する情報（何が、何を、何について、何に関する）が含まれていない。そこでまず、意味が「ある」情報を含む文を調べ、考察した。

- SUN の F77 で C のソースをリンクする方法を教えて下さい。→ **Question**
- mailx の日本語化されたもの、できれば、MIME にも対応したものがあれば、教えて下さい。 → **Question**
- SUN SS20 をリブートしたところ、OPENWINDOW が立ち上がらなくなりました。→ **Fail**

これらのセンテンスは、「何が／を／について」に相当する部分がきちんと明示されている。この「何」の部分には、名詞的概念を表す語が来る。この名詞的概念が表す語には、２種類のものが考えられる。１つはコンピュータ関連の専門用語（固有名詞）である。コンピュータ関連の専門用語は、ほとんどアルファベットかカタカナで記述される語である。このような語は、形態素解析システム JUMAN3.5[4] において未定義語と解析される。そこで、未定義語が文中に存在すれば、意味ありセンテンスと判定することにする。

もう１つは、一般的に使われる名詞（「方法」、「本」など）である。これらの名詞を網羅することは、現実的ではない。そこで、意味なしセンテンスに現れる名詞をリストアップし、このリストに含まれない名詞を、意味のある名詞とみなす方法を採用する。

以上の考えに基づき、以下のアルゴリズムで意味なしセンテンスの判定を行なう。

意味なしセンテンス判定アルゴリズム

1. 入力文を JUMAN により形態素解析する。
2. 文中に未定義語が、
   - ある → 意味ありセンテンス
   - ない → 「意味なしセンテンス候補」 → 3. へ
3. 「意味なしセンテンス候補」中の名詞が、「意味なし名詞リスト」に、
   - 含まれないものがある → 意味ありセンテンス
   - すべて含まれる → 「意味なしセンテンス」

本アルゴリズムで意味なしセンテンスと判定される文の例を図 3.2 に示す。

- どなたか解決方法などご存知の方がいらっしゃいましたら御教授下さい。
- 一体、なにが悪いのでしょうか？
- 私の分かる範囲で調べて見ましたが分かりません。

図 3.2: 意味なしセンテンスと判定される文の例

3.3.3 意味なし名詞リストの作成

前項で用いる「意味なし名詞リスト」を以下の方法で作成した。

1. 1998 年 1 月から 12 月までの記事に対して、文のタイプ判定を行ない、タイプが **Goal**、**Fail**、**Question** の文を抽出した。
2. 各々の文を形態素解析し、未定義語を含む文を除去した。
3. 残った文に含まれるサ変名詞以外を抽出し、これを意味なし名詞リストとした。

サ変名詞は、「〜する」をつけると動詞的な意味を持つ名詞なので、意味なし名詞リストには加えない。上記の方法で得られた意味なし名詞リストを図 3.3 に示す。

えま かた かたはい から手 かんじ くらい ぐらい こと ごぞんじ ご存じ ご存知 さい しゃれ すか ずれ せん そのた ため だれ とき ところ ど どなた な なに の のかた は ば ほお ほか みなさま も もの やり方 よう らしゃ 案 以下 以外 以上 以前 異常 一夜 引 英 英語 下 何 何処 何方 価格 価値 画面 回路 皆さん 学割 環境 間違い 願 機械 気 気持 記事 泣き 教え 興味 型 結 件 見当 源 現象 言語 限り 個人 後 御心 御存じ 懇親会 際 策 錯誤 参考 参考書 仕方 仕様 市場 私 資源 資料 事 事柄 字 持ち 時間 自動 自分 若松 手 手順 手段 需要 初め 初心者 所 商用 症状 上 上記 場合 場面 情報 状況 状態 色 心 心当たり 図 正解 節 先度 先日 線 前 素人 相場 装 送料 速度 他 他人 対策 代理店 誰 端子 値 知恵 潮流 定 的 点 電源 等 頭 読み物 内容 軟弱 日本語 版 範囲 番 必要 貧乏性 不成功 分かり 分岐点 文字 別 辺 辺り 方 方々 方策 方法 法 本 目安 問題 問題点 優 勇者 余分 様 要因 落し穴 理由 量

図 3.3: 意味なし名詞リスト

3.4 長さの異なる重要文の抽出

重要文を抽出し、2 種類の長さの要約を生成する。ひとつは短めの要約であり、もうひとつは、長めの要約である。

3.4.1 短めの要約の抽出

短めの要約は以下の方法で抽出する。

	センテンス	タイプ
1	こんにちは、片原@シャドウ・エンターテインメント、です。	(挨拶)
2	Solaris 2.5 for x86 で使用できる LAN ボードについて教えて下さい。	(Ques, 環境)
3	最初はどこの製品か分からないボード（Win95 では NE2000 互換）を接続していたのですが、OS インストール時にネットワークの設定画面を表示しませんでした。	(Fail)
4	・インストール終了後「ifconfig -a」でもインターフェイスは表示無し。	(その他)
5	そこで MELCO 社の LGY-VI-T に変えると設定はできるのですが、ネットワークへは接続できませんでした。	(Fail)
6	（MELCO のサポートは Solaris をサポートしてないので答えられないと言った）	(環境)
7	・ネットマスク 0xffffffe0、NIS+ 他は未使用、で設定・「ifconfig -a」ではインターフェイスは動作している様に見える。	(その他)
8	Solaris2.5 で LAN ボードを接続するには特別な設定が必要なのでしょうか？	(Ques)
9	よろしくお願いします。	(終挨拶)

図 3.4: 記事全文

- 文のタイプが **Goal**、**Fail**、**Question** のいずれかであり、かつ、意味なしセンテンスではない文のうち、記事の最初に現れる文を抽出する。

図 3.4に示した記事からの抽出例を図 3.5に示す。

3.4.2 長めの要約の抽出

長めの要約は以下の方法で抽出する。

- 文のタイプが **Goal**、**Fail**、**Question** のいずれかであり、かつ、意味なしセンテンスではない文を全て抽出する。

図 3.4に示した記事からの抽出例を図 3.6に示す。

3.5 従来研究との比較

本研究の要約生成の方法は、先の SUN QA-Pack[2] における要約生成の方法を改良したものとなっている。SUN QA-Pack の要約生成では、手がかり表現を用いて、抽出すべ

	センテンス	タイプ
2	Solaris 2.5 for x86 で使用できる LAN ボードについて教えて下さい。	(Ques, 環境)

図 3.5: 短めの要約

	センテンス	タイプ
2	Solaris 2.5 for x86 で使用できる LAN ボードについて教えて下さい。	(Ques, 環境)
3	最初はどこの製品か分からないボード（Win95 では NE2000 互換）を接続していたのですが、OS インストール時にネットワークの設定画面を表示しませんでした。	(Fail)
5	そこで MELCO 社の LGY-VI-T に変えると設定はできるのですが、ネットワークへは接続できませんでした。	(Fail)
8	Solaris2.5 で LAN ボードを接続するには特別な設定が必要なのでしょうか？	(Ques)

図 3.6: 長めの要約

き重要文を判定していたが、本研究では、手がかり表現を文のタイプを決定するものとして整理し直した。さらに、先の研究では、１、２文からなる１種類の要約を生成する方法となっていたが、本研究では、１文からなる短めの要約と、数文からなる長めの要約を生成することを実現した。１文からなる短めの要約は、先の研究が生成する要約とほぼ対応する。この要約は、その質問記事を読むべきか否かを判定するスクリーニングのために有効であると考えられる。一方、本研究ではじめて実現した長めの要約は、質問者の目標 (**Goal** 文)、問題点（**Fail** 文)、質問 (**Question** 文）をすべて含んでいるので、その質問記事を読まなくても、ほぼその内容を把握することが可能となると考えられる。

第 4 章

実験と検討

本章では、3 章で説明した要約生成システムのタイプ判定モジュールと要約生成モジュールの評価実験を行ない、その結果を検討する。

4.1 タイプ判定の評価

タイプ判定モジュールは、1997 年 1 月の 90 記事をもとに作成した。これとは異なる 1998 年 1 月の 59 記事（未知データ）を対象に、その記事に現れる文のタイプ判定を行なった。ここでは、要約として抽出される文のタイプ **Goal**、**Fail**、**Question** に対する評価を行なった。表 4.1にその結果を示す。

表 4.1: **Goal,Fail,Question** の判定結果

	文数
(1) Goal,Fail,Question と判定されたもの	134
(そのうち意味なしセンテンスとみなすべきもの)	(19)
(2) Goal,Fail,Question と判定すべき文で、そう判定されなかったもの	14
タイプ G,F,Q と判定されるべき文の総数 (1)+(2)	148

Goal、**Fail**、**Question** と判定すべき文で、そう判定することができなかった 14 文の原因は、必要なパターンが文字列パターンの定義に入っていなかったためである。語幹は定義されていても、語尾が違っているものがほとんどであった。

意味なしセンテンスは、次のステップで排除するので、タイプ判定モジュールでは、

Goal、**Fail**、**Question** と判定されてもよいと考えると、これら３タイプ判定の正解率は以下のようになる。

$$\frac{134}{148} = 91\%$$

以上により、重要文抽出に必要な３つのタイプは、高い精度で判定できることが明らかになった。

4.2 要約生成の評価

1997 年 3 月の未知記事（要約生成システムを作成する際に調査した 90 記事ではない、新たに入手した記事）45 記事に対して、要約を生成する実験を行なった。その結果を、以下の３つの基準で評価した。

Good 抽出された文は要約として適切である。

OK 抽出された文は要約として適切ではないが許容できる範囲にある。

No Good 抽出された文は要約として許容できない。あるいは全く抽出できていない。

短めの要約の評価結果と長めの要約の評価結果を表 4.2 に示す。

表 4.2: 要約の評価結果

	短めの要約		長めの要約	
	記事数	割合 (%)	記事数	割合 (%)
Good	36	80	35	78
OK	5	11	9	20
No Good	4	9	1	2
Total	45	100	45	100

要約として成立する範囲（Good ＋ OK）の要約精度は、

短めの要約の場合　91%
長めの要約の場合　98%

となった。ネットニュースの記事は、新聞や論文などに比べ、誤字や脱字、文法的誤りが多いことを考慮すると、高い精度であるといえる。但し、対象とした記事数が少ないので、より多くの記事に対して適用し、確かめる必要がある。

要約として不適切なものを抽出する原因には、以下の原因がある。

- 文に特徴的な表現が使われていないため、重要文が判定されなかった。
- 記事の主題が fj.sys.sun の内容とずれている、あるいは全く関係ない記事であった。

前者は、タイプ判定モジュールの精度を上げることによって解決できると考えられる。後者は、本システムの範囲外の問題である。

4.3 関連研究

テキストからその要約を自動生成する方法は、古くから研究されてきている [1, 5]。その中心的な方法は、テキストのそれぞれの文に対して重要度を計算し、その重要度の高い文を抽出するという、重要文抽出法である。

本研究の要約手法も、この重要文抽出法に分類される。本研究の対象テキストである fj.sys.sun の質問記事は、誤りを多く含んだ品質の低いテキストであるため、広く用いられている重要文抽出法は有効に機能しないと考えられる。そのため、本研究で提案した要約手法を用いた。本手法の特徴は、以下の通りである。

- 対象テキストを fj.sys.sun の質問記事に限定している。
- 対象テキストの品質が低いため、形態素解析を利用せず、文字列のパターンマッチングを用いている。
- 質問記事固有の表現に注目し、文のタイプを判定する。重要文は、このタイプに基づいて決定する。すなわち、文に対して重要度（数値）を計算しない。
- 短めの要約と長めの要約の２種類の要約を生成する。

第 5 章

結論

本研究では、２種類の長さの異なる要約を生成するシステムを作成した。一つは１文のみからなる短めの要約であり、もう一つは数文で構成される長めの要約である。

長さの異なる要約の生成を実現するため、本研究の研究対象であるニュースグループ fj.sys.sun の調査を行なった。質問記事中でよく現れる特徴的な表層表現を用い、文のタイプを次の１０種類設定した。

1. 挨拶、自己紹介，2. 環境，3.Goal，4.Fail，5.Question，
6.Error または Program の表示，7. 分析，8. 終りの挨拶，9.Signature，10. その他

fj.sys.sun の質問記事において中心となる情報は、「何を行ないたいのか」、「直面した問題は何か」、「質問は何か」ということである。「何を行ないたいのか」を記述した文は、上記のタイプの **Goal** となる。このタイプの文では、意志を表す助動詞「〜たい」がよく用いられる。「直面した問題は何か」を記述した文は、上記のタイプの **Fail** となる。このタイプの文では、否定文であるか、あるいは、文末に否定的な表現を表す動詞が使われることが多い。「質問は何か」を記述した文は、上記のタイプの **Question** となる。このタイプの文では、ほとんどの場合、疑問文となる。これら３つのタイプを fj.sys.sun の重要文として抽出した。

要約の生成は以下の手順で行なった。まず質問記事中の各々の文が、どのタイプ文であるかを、文字列のパターンマッチングによって判定した。その際、意味のある情報を全く含まない文が、抽出すべき重要文（**Goal**,**Fail**,**Question**）と判定される場合がある。それらは重要文として抽出するのは不適切である。本論文では、このような情報を全く含まない文を意味なしセンテンスと定義した。

意味なしセンテンスの特徴として、「何が／を／について」という対象を特定する名詞的概念を表す語が、センテンス中に含まれていない。これら名詞的概念を表す語には２種

類考えられる。１つはコンピュータ関連の専門用語（固有名詞）である。コンピュータ関連の専門用語は、ほとんどアルファベットかカタカナで記述される語であるため、簡単に判定できる。もう一つは、一般的に使われる名詞（「方法」、「本」など）である。これらの名詞を網羅することは、現実的ではない。そこで、意味なしセンテンスに現れる名詞をリストアップ（意味なし名詞リスト）し、このリストに含まれない名詞を、意味のある名詞とみなす方法を採用した。従って、重要文と判定された文中の名詞が、コンピュータ関連の専門用語を含まず、かつ、意味なし名詞リストに全て含まれる場合、その文を意味なしセンテンスと判定した。

最後に、長さの異なる２種類の要約を生成した。短めの要約は、文が重要文と判定され、かつ、意味なしセンテンスではない文のうち、記事の最初に現れる文を抽出した。長めの要約は、文が重要文と判定され、かつ、意味なしセンテンスではない文を全て抽出した。

本システムを未知の記事に対して適用したところ、短めの要約の場合 30 記事中 26 記事（87%）に対して、長めの要約の場合 30 記事中 28 記事（93%） に対して、適切な要約または許容できる要約を生成することができた。

謝辞

本研究を進めるにあたり、終始熱心な御指導、励ましを賜わりました佐藤理史助教授に、心から感謝致します。

また、本論文を作成するにあたって、佐藤円博士に多大な御協力を頂き、厚く御礼申し上げます。

最後に、多くのアドバイスと御協力を頂いた知識工学講座の皆様に深く感謝致します。

参考文献

[1] 奥村学, 難波英嗣: テキスト自動要約技術の現状と課題. JAIST Research Report, IS–RR–98–0010I, 北陸先端科学技術大学院大学情報科学研究科, 1998.

[2] 佐藤円, 佐藤理史: ネットニュース記事群の自動パッケージ化. 情報処理学会論文誌, Vol.38, No.7, pp.1225–1234, 1997.

[3] Donald A.Norman, 野島久雄（訳）: 誰のためのデザイン？. 新曜社, 1990.

[4] 黒橋禎夫, 長尾真: 日本語形態素解析システム JUMAN version 3.5. 京都大学大学院工学研究科, 1998.

[5] 佐藤理史, 奥村学: 電脳文章要約術 — 計算機はいかにしてテキストを要約するか —. 情報処理, Vol. 40, No. 2, 1999, (掲載予定).