構文・意味保存による多言語 Web ページの作成支援システム

浦江　宏志†　手塚　太郎‡　木村　文則‡　前田　亮‡

†立命館大学 理工学研究科 〒525-8577 滋賀県草津市野路東 1-1-1
‡立命館大学 情報理工学部 〒525-8577 滋賀県草津市野路東 1-1-1

E-mail: †cm002067@ed.ritsumei.ac.jp, ‡{ tezuka,amaeda }@media.ritsumei.ac.jp, fkimura@is.ritsumei.ac.jp

あらまし Web ページの多言語化の方法としては，ページ作成者による事前の翻訳，ならびにページ閲覧者による閲覧時の翻訳サービスの利用が挙げられる. 前者は閲覧者にとって読みやすい文章になるという利点があるが，ページ作成者にとっては大きな負担となる．一方後者は，ページ作成時の負担はないが，機械翻訳で行われるため，ページ作成者の意図とは違う意味で翻訳されてしまうことがある．そこで本研究では，ページの作成時に文の構造と意味を保存することで，翻訳時に作成者の意図を適切に反映させる手法を提案する．また，構文と意味の保存を自動化し，ページ作成者が保存結果を編集できるようにすることで， ページ作成者の負担をさらに少なくする手法についても提案を行う.

キーワード Web，多言語化，翻訳，セマンティック Web

Structural and Semantic Indexing for Supporting Creation of Multilingual Web Pages

Hiroshi URAE†　Taro TEZUKA‡　Fuminori KIMURA‡　and　Akira MAEDA‡

†Graduate School of Science and Engineering, Ritsumeikan University
1-1-1 Noji-higashi, Kusatsu, Shiga 525-8577, Japan

‡College of Information Science and Engineering, Ritsumeikan University
1-1-1 Noji-higashi, Kusatsu, Shiga 525-8577, Japan

E-mail: †cm002067@ed.ritsumei.ac.jp, ‡{ tezuka,amaeda }@media.ritsumei.ac.jp, fkimura@is.ritsumei.ac.jp

Abstract There are two ways of translating web pages. One is preparing translated web pages made by the webmaster. Another is using web translation services. The former way has a merit that translated web pages consist of natural sentence. It imposes, however, a burden on webmaster. The latter way has a merit that it doesn't impose a burden on the webmaster. It often makes, however, unnatural sentences the webmaster doesn't intend. In this paper, we propose a new method that makes natural sentences by using an analysis of sentence structures and what each word means. This system lightens a burden on the webmaster analyzing sentence structures and what each word means almost automatically.

Keyword Web, Multilingualization, Translation, Semantic Web

1. はじめに

World Wide Web は世界中の情報へのアクセスを可能にしたが，コンテンツの記述に使用される自然言語の多様性が情報流通における大きな障壁として残っている．この障壁を乗り越えるため，様々な形で Web ページの多言語化が行われている.その方法として主に，Web ページの作成者が独自の方法で多言語化を行う方法と翻訳サービスを利用する方法が挙げられる.

前者では，ユーザの使用言語に応じて，Web ページを言語ごとに用意したり，PHP や JavaScript といったスクリプト言語を用いて Web ページの内容を書き換えることによって，実装している．この方法では，Web ページの作成者が予め各言語で書かれた文章を用意しているため，ユーザにとって自然で読みやすい文章であることがメリットである．しかし，Web ページの内容を更新したり，新たな Web ページを作成する度に，各言語に対応した文章を用意しなくてはならないことがデメリットである．このデメリットは対象とする言語が多いほど増大するため，Web ページの作成者に対して大きな負担となる.

後者では，Web ページの閲覧時にユーザが Google 翻訳や Yahoo!翻訳などの翻訳 Web サービスを用いることで実現している．この方法では，Web ページの作

成者が Web ページの多言語化を実装していない場合でも，ユーザは自分の使用言語で Web ページを閲覧することができる．しかし，翻訳精度には限界があるため，翻訳結果が不適切であることも多い．これは，現在の機械翻訳では Web ページ作成者の意図を汲み取れない場合があり，文の構造や単語の意味を取り違えていることが一因である．

そこで我々はこれまでに，Web ページの作成段階で文の構文および単語の意味を指定することにより，Web ページの多言語化を支援する手法を提案した[1]．この手法は Web ページ作成時に行う，文章がどのような構造になっているかを保存する「構文保存」と各単語の意味を保存する「意味保存」，Web ページの閲覧時に行う「文章復元」の 3 段階からなる．構文保存・意味保存をシステムによって自動で行うことで，Web ページの作成者が多言語化を行う際にかかる負担を少なくすることができる．その一方で，最終的な構文・意味保存の結果を Web ページ作成者が確認，編集することを可能にすることで，Web ページ作成者の意図を正確に反映させることができる．これによって閲覧時に翻訳サービスを利用する手法よりも高い精度が期待できる．本論文では，構文保存，意味保存の詳細な表現方法について述べる．さらに，文書復元の手法を提案し，システムの実装について述べる．

2. 関連研究

近年，統計処理手法による言語解析の限界が指摘され，統計処理手法と深い言語解析を組み合わせた手法が注目されている．深い言語解析の解釈は様々であるが，増市らは，「文の構成要素間の修飾関係だけでなく，述語・項構造まで特定する処理」とし，深い言語処理による，複数言語の文法記述および，文の解析生成システムの研究を行った[2]．この研究で使われている深い言語処理の為の言語理論は，Lexical Functional Grammar(LFG)[3]と呼ばれる．この理論では，自然言語文の構造を 2 つの構造で表している．1 つは木構造で表した c-structure，もう 1 つは文の格構造や時制などの意味情報をマトリックス構造で表現した f-structure である．c-structure は，言語毎に大きく異なるが，f-structure は異なる言語間でも違いが少ないことがわかっている．本研究では，複数の言語間で統一的な文の構造の表現が必要であるため f-structure のように文の構造を文法的機能で表す．

3. 提案手法

システムの概要図を図 1 に示す．

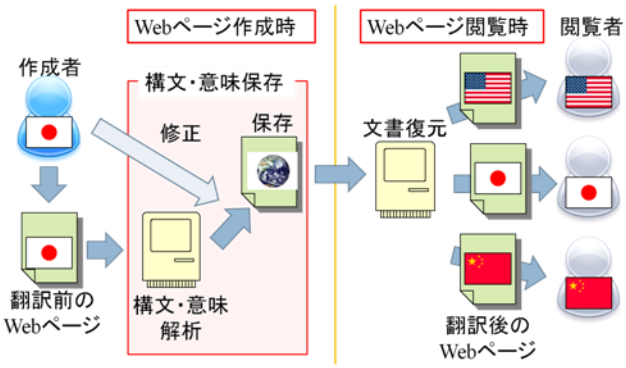

図 1　システム概要図

3.1. 構文保存

構文保存は，文章がどのような構造であるかを保存する処理である．この処理では，構文解析器や形態素解析器を用いて文章を解析し，主語（S）や述語（V），目的語（O）といった基本的な構造（以下，基本構文）とそれに係る修飾語（M）に分解し，文章の構造を保存する．

ここで「He works for a local bank.」という英語の文章と，日本語の対訳となる「彼は地元の銀行に勤めている。」という 2 つの文を例にあげる．異なる言語では，図 2 のように語の順序も異なる．その為，構文保存では語の順序は保存しない．

まず，「He works for a local bank.」という英語の文章の解析結果は図 3 のようになる．文章を構成していた各語は元の順序を保持せず，基本構文と修飾語によって表す．この時，修飾語にはどこに係っているかという情報を付加する．また，各語は基本形で保存し，必要に応じて時制情報を付加する．同様に，「彼は地元の銀行に勤めている。」という日本語の解析結果は図 4 のようになる．ここで，英語の文章「He works for a local bank.」と日本語の文章「彼は地元の銀行に勤めている。」の解析結果に注目すると，元の文の関係を表す図 2 では対訳関係である語同士が異なる順序で並んでいたのに対し，文章の構造を解析した後では，図 5 のように対訳関係にある語同士を，同じ構造で表せている．

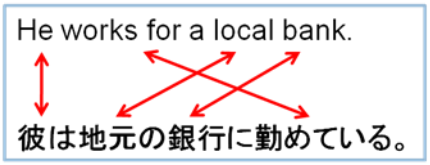

図 2　異なる言語で対訳関係にある文章の語の並びの違い

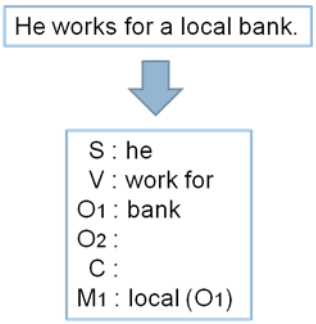

図 3　「He works for a local bank.」の構文保存の例

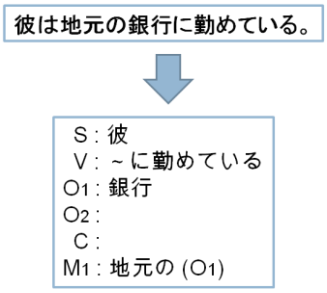

図 4　「彼は地元の銀行に勤めている。」の構文保存の例

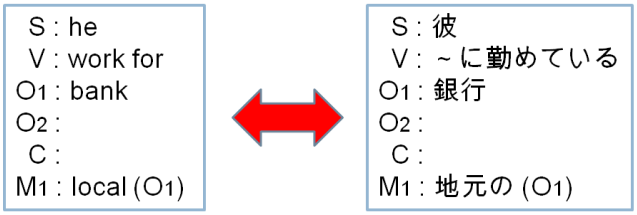

図 5　異なる言語で対訳関係にある文章の解析結果の比較

3.2. 意味保存

意味保存では，形態素解析した結果をもとに，各語の意味とその意味 ID を格納しているワード ID データベース（表 1）から参照し，意味を保存する．ワード ID データベースは英和・和英対訳辞書「英辞郎」を用いて作成した．単語によっては複数の意味を持つこともあるが，意味ごとに別の ID を割り振り，ワード ID データベースに格納する．また，熟語に関しても同様に意味 ID を付与してある．このワード ID データベースを基に，各語の意味は意味 ID によって保存される．

Web ページの作成者の母国語によって作成された文章は，構文保存と意味保存の処理によって，意味 ID と構文のみの特定の言語に依存しない形式に変換される．

例として，図 3 の構文保存の処理結果に対して意味保存の処理を行ったのが図 6 である．「bank」には「銀行」や「岸」という意味が存在する．このような曖昧性のある語に対して，システムは曖昧性のある語が文章中に含まれることを表示する．Web ページの作成者はその語にふさわしくない意味 ID が付与されている場合，手動で変更することができる．また，「work for」は 1 つの熟語として保存する．これにより 2 つのメリットが得られる．1 つ目のメリットは曖昧性の減少である．ワード ID データベースには現在，「work」に対して 27 の意味，「for」に対して 10 の意味が保存されている．ここから推測される「work for」の意味の候補は 270 となる．この中から意味を選ぶことは Web ページの作成者にとって大きな負担となる．しかし，「work for」という熟語の意味をワード ID データベースに登録しておくことにより，Web ページの作成者は実際に登録されている 6 つの意味から選ぶことができ，大きく負担を減らすことができる．2 つ目のメリットは特殊な意味への対応である．熟語は構成要素となる各語の意味の組み合わせでは表せない特殊な意味を持つことがあるため，熟語として保存することで対応している．

図 4 の構文保存の結果に対しても同様に処理を行った結果が図 7 である．対訳関係にある 2 つの文章「He works for a local bank.」と「彼は地元の銀行に勤めている。」に対して構文保存と意味保存の処理を行った結果，全く同じ形になっていることから，同じ内容を意味する複数の言語の文章を一つの形式で表せることが示されている．

しかし，現状では曖昧語の自動判定を行うことが難しく，Web ページの作成者にかかる曖昧語判定の負担が大きい．その為，Web ページの作成者の負担をより少なくするために，システムがより多くの語を自動で判定できるようにする仕組みが必要である．解決案の一つとして，図 8 のように意味 ID を概念ごとに分け，その文書内でより多く使われている概念の意味で意味保存を行うという手法が考えられる．

表 1　ワード ID データベース

意味 ID	en	ja
157833	bank	土手 岸
157844	bank	～を積み上げる ～を山にする
157850	bank	銀行
824570	he	彼
1068632	local	地元の 特定の場所の 現地の その地域の 地場の
2042928	work for	～に勤めている

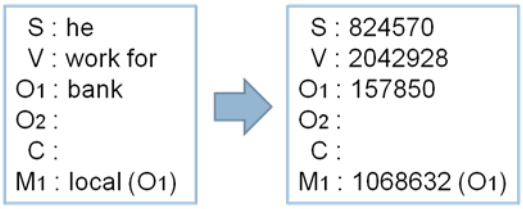

図 6　「He works for a local bank.」の
意味保存の例

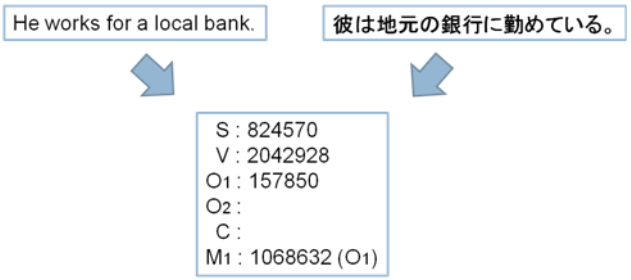

図 7　「He works for a local bank.」と
「彼は地元の銀行に勤めている。」の構文保存，
意味保存結果の比較

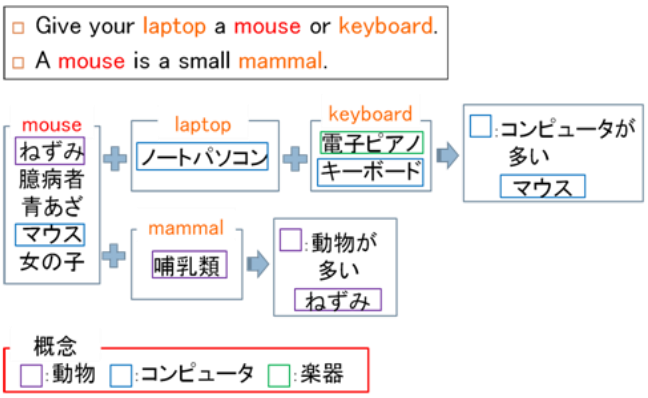

図 8　概念を用いた曖昧性の解消

3.3. 文章復元

構文と意味を保存した文書にユーザがアクセスした時，そのユーザの使用言語に合わせて文章を復元する．復元は保存と逆の順番で処理する．まず，ユーザの使用言語を取得する．その情報をもとにワード ID データベースを参照し意味を復元し，その後基本構文を基に構文を復元し修飾語を付け加える．

図 7 のように保存された文章を英語を使うユーザがアクセスした場合の文書復元例が図 9 である．この例では英語を母国語とするユーザがアクセスしたと想定している．ワード ID データベースを参照し，各語の意味 ID を英語での単語に復元する．その後，基本構文を英語の文章に復元し，最後に修飾語を付け足す．

文章復元の課題点として文章の構成の曖昧さが課題となる．英語のように基本構文が確立している言語は比較的簡単に復元できると考えられるが，日本語のように文の構造が曖昧な言語では，復元のパターンを見つけられるかが課題となる．その解決策として，基本構文を機械翻訳し，それに修飾情報を付け足していくという手法が考えられる．

また，「a」や「the」といった冠詞，三人称単数現在形や過去形といった時制，「は」や「が」といった格など，各言語に大きく依存する部分をどのようにして自然な文章となるように復元するかも課題となる．

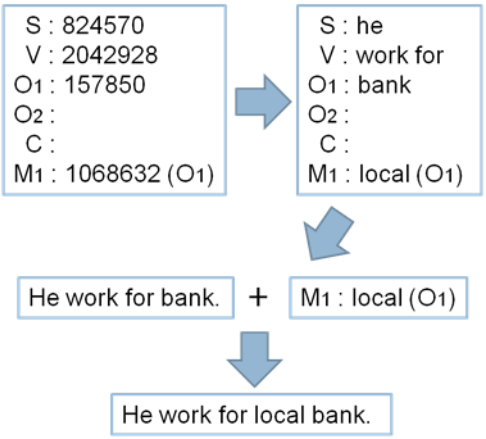

図 9　文書復元の例

4. システム実装

本章では「Bank is a raised portion of seabed or sloping ground along the edge of stream, river or lake」という文を例とし，手動による処理を交えながら，構文保存，意味保存および文書復元を行う例を説明する．この処理を行うに当たって，文章の作成者は英語を母国語とし，閲覧者は日本語を母国語とすると想定する．この文を Google 翻訳を用いて日本語に翻訳した場合，「銀行が流れ、川や湖の縁に沿って海底や傾斜地の隆起部です。」という結果が得られる。Google 翻訳による翻訳結果をシステムによる処理の比較対象とする．

まず，構文解析を行う．構文解析には APP(Apple Pie Parser)[4]を用いた．解析結果は図 10 である．この解析では，基本構文は SVC となり，C は「a raised portion」を「of seabed or sloping ground」が修飾した名詞句である．さらに，C に対し M「along the edge of a stream, river or lake」が修飾としてかかっている．

次に，構文の修正を手動で行う．APP による解析では「a raised portion of seabed or slopping ground」と「along the edge of a stream, river or lake」に分かれる．しかし，正確には「a raised portion of seabed」と「sloping ground along the edge of stream, river or lake」が「or」によって並列関係となっているとなるべきである．そこで，C と M の関係を修正し，図 11 のように 2 つの名詞句の並列関係全体が C となるように修正し，構文情報として保存する．

続いて，ワード ID データベースを参照し，各語に対し意味 ID の付与を行う．その結果が図 12 である．

次に，意味の修正を手動で行う．「Bank」という語は「銀行」意味する ID「157850」が付与されたが，この文での「Bank」は「土手」を意味する．そこで、ワード ID データベースの「Bank」の他の意味候補を参照し，図 13 のように「土手」を意味する ID「157833」に修正し，この結果を意味情報として保存する．

最後に，この文章を日本語に復元する．復元には Google 翻訳を用い，日本語の文章では各語がどのような構造をとるのかという情報を得る．長い文章をそのまま翻訳すると，誤訳や間違った構造の文が得られてしまう．しかし，文章が短く，単純であれば、ある程度機械翻訳の精度は良くなる．そこで，修飾情報を伴わない基本文「Bank is a raised portion or sloping ground」のみを翻訳し，その結果に対して修飾語を補うという手法を提案する．この結果,「銀行は、隆起部や傾斜地です。」という文章が得られる．しかし，この結果では「銀行」という誤訳が含まれているせいで,「土手」を意味する ID「157833」がどこに位置すべきなのかが判断できない．そこで文章をさらに抽象化し，S 全体を「S」，全体を「C」という 1 文字で置き換え，翻訳を行う．この結果「S は C です」という文章が得られる．よって日本語では，SCV という文の構造になるとわかり，これからワード ID データベースを元に基本文の意味を復元すると「土手高くした部分か傾斜した地面です」となる．さらに各語に対してそれぞれの修飾語を付与すると，最終的な結果として「土手海底の高くした部分か小川、川か湖の端に沿って傾斜した地面です。」という翻訳結果を得られる．この文章は Google 翻訳に比べ，作成者の意図をより自然で正しく反映した翻訳になっていると言える．

S : Bank
V : is
C1 : a raised portion
C2 : of (C1)
C3 : seabed or sloping ground (C2)
M1 : along (C)
M2 : the edge (M1)
M3 : of (M2)
M4 : a stream, river or lake (M3)

図 10　APP による構文解析結果

S : Bank
V : is
C1 : a raised portion
C2 : of (C1)
C3 : seabed (C2)
C4 : or
C5 : sloping ground
C6 : along (C5)
C7 : the edge (C6)
C8 : of (C7)
C9 : a stream, river or lake (C8)

図 11　構文修正結果

S : 157850
V : 166701
C1 : 1503101 + 1411693
C2 : 1275349 (C1)
C3 : 1622678 (C2)
C4 : 1300566
C5 : 1695777 + 783652
C6 : 65821 (C5)
C7 : 547633 (C6)
C8 : 1275349 (C7)
C9 : 1767312 + 1584292
　　+ 1300566 + 1019154 (C8)

図 12　意味 ID 付与結果

S : 157833
V : 166701
C1 : 1503101 + 1411693
C2 : 1275349 (C1)
C3 : 1622678 (C2)
C4 : 1300566
C5 : 1695777 + 783652
C6 : 65821 (C5)
C7 : 547633 (C6)
C8 : 1275349 (C7)
C9 : 1767312 + 1584292
　　+ 1300566 + 1019154 (C8)

図 13　意味 ID 修正結果

5. おわりに

本論文では，Web ページの作成段階で文の構文および単語の意味を指定することにより，Web ページの多言語化を支援する手法を提案した．

しかし，本手法を実装した場合のメリットは Web ページの多言語化の実装を補助するだけではない．Web ページそのものが特定の言語に依存しない形で保存されているため，言語横断検索が可能である．例えば「銀行」に関する文書を複数の言語で書かれた文書群から探す場合，日本語なら「銀行」，英語なら「bank」，イタリア語なら「banca」，フランス語なら「banque」と，対象とする言語数だけ検索クエリが必要となる．一方，本手法で必要なクエリは 1 つである．なぜなら，もともと「銀行」と記述されていた場合も，「bank」と記述されていた場合も，意味保存の段階で同じイン

デックス，表 1 のワード ID データベースを使用したとすると「157850」に書きかえられて保存されているためである.この結果,ユーザの母国語のクエリから，各言語で「銀行」を意味する意味 ID「157850」に変換し検索するだけで，言語横断検索が可能となる.

さらに,本手法は Web ページ以外の多言語化への応用も可能である．メールや論文など，作成者と閲覧者の使用言語が異なる場合がある文書に対しては，Web ページ同様に有用であると推測される.

また，構文を保存することで構文情報を考慮した検索も可能となる．今までは検索クエリが文章中でどのような使われ方をしているかを指定することができなかった．しかし本手法では，検索クエリが主語として使われている文章を探すといった検索も可能である．さらに，2 語以上の検索クエリを使用して検索を行う場合に，語と語の関係性を指定した検索方法も考えられる．これにより，質問応答システムなどの自然言語処理の分野においても応用が期待される.

その他にも,構文情報や意味情報を利用できるようになることから，コンピュータが Web ページの内容を解析できるセマンティック Web の実現への寄与も期待できる．この応用をより有効に利用するために，データベースに意味 ID とそれに対応する各言語での単語の他に，その語が何を表す語なのかをメタデータとして付与すると,より良い結果が得られると考えられる.

参 考 文 献

[1] 浦江 宏志，手塚 太郎，木村文則，前田 亮，構文・意味保存による多言語 Web ページの作成，第 18 回 Web インテリジェンスとインタラクション研究会, pp.13-14, 2010-9.

[2] 増市博，大熊智子，鷹合基行，Lexical Functional Grammar に基づく言語解析の現状とその応用，電子情報通信学会技術研究報告，NLC，言語理解とコミュニケーション 106(299), 1-8, 2006-10-13.

[3] R. M. Kaplan, and J. Bresnan, Lexical-Functional Grammar: A formal system for grammatical representation, in The Mental Representation of Grammatical Relations, pp.173-281, The MIT press, 1982.

[4] Proteus Project - Apple Pie Parser (Corpus based Parser), http://nlp.cs.nyu.edu/app/, 2011-2-14