

状況と機能に応じた日本語の学習を支援するシステム

劉 軼 加藤 伸隆 馬目 知徳 伊丹 誠 伊藤 紘二

東京理科大学 基礎工学部

1 はじめに

我々は、学習者に状況と機能に応じた日本語表現を広く体系的に学習させるために、編集者の用意する状況・機能手がかりが学習者と誤り診断パーザの双方に制約として用いられるシステムを提案している [1]。

本システムは、対話例のデータベースを持ち、その各々について、状況・機能的に重要な部分を空白部として、その部分の状況と機能を手がかりインデックスとして編集しておき、学習者にはそれ（一般にはその一部）を伝えた上で、作文を行なわせる。学習者は、自立語については提示された語彙を用い、活用と機能的語彙を考えて補い、入力文を作成する。システムは、作成された入力文を、誤り診断機能をつけた汎用のパーザによって解析し、構文・意味的に誤りがあったり、その時の状況にそぐわなかったりした場合には、誤答部分の表層表現を持つ例文あるいは正解に対応する手がかりインデックスをもつ例文を対話例データベースから検索して学習者に提示し、その手がかりを空白部の手がかりと比較させ、学習者の答と対話例の表現とを比較させた上で、学習者に改めて作文させる。これらのガイダンスにもかかわらず正解に至らない場合は、誤答に最も近い正解とのずれについてバグ仮説に基づいたコメントを提示する。

2 日本語学習支援システムの構築

システムの構成は、まず、データモジュールとしては、穴埋め問題を含む対話例のデータベース、穴埋め入力の誤りを診断するパーザの為の辞書とバグ仮説（現状では、パーザに組み込み）、対話例に含まれる重要表現につけられて、検索と診断の手がかりとして用いられる状況・機能インデックス、そして、状況・機能インデックスの素材としてのプリミティブのレポトリがある。これらの内基本的なものは、システム提供側が提供するが、対話例データベースと状況・機能インデックスは教師が編集して、追加してゆくことができるようにエディタを整備する必要がある。手続きモジュールとしては、インターフェースのほか、対話

例検索提示、誤り診断、コメント生成と、学習支援の流れを作る学習管理モジュールがある。

学習支援は、図1に示すように、次の流れに従うものとする。

- (1) 教師／学習者が選んだ対話例をシステムが提示し、状況を媒介語で説明する。
- (2) テキストに現れる単語については、学習者が要求すれば、媒介語訳を提示する。
- (3) 対話例の空白部分について、手がかりインデックス（一般にはその一部）を媒介語使用のフレーム表現で提示した上で、作文入力をさせる。作文については、自立語は提示リストから引かせ、活用と機能的語彙についてはキーボードあるいはメニューから入力させる。
- (4) 穴埋め作文入力を誤り診断パーザで解析する。

(4a) 解析の結果、正解である場合:

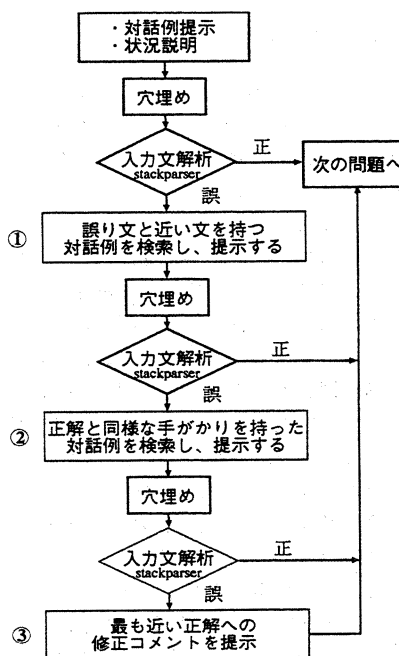


図1: 本システムにおける学習支援の流れ

次の問題へ。なお、関連した表現について学習者から要請があれば、状況・機能インデックスを用いることにより検索して、提示する。

(4b) 解析の結果、誤りが見つかった場合:

学習者に対し、誤った部分を気づかせる為のヒントとして、学習者の入力に最も近い正解とずれている表層表現部分を含んだ対話例をシステムが検索して提示し、状況・機能インデックスによって状況・機能の違いを比較させる。

(5) ここで、再度、解答を入力させ、パーザで解析する。

(5a) 結果が正解の場合 (4a) へ。

(5b) 再度誤った場合、穴を埋める正しい表現に気づかせる為のヒントとして、自立語の相異を除き穴埋め部分と同一の状況・機能インデックスを持つ対話例を検索して提示する。

(6) ここで、再度解答を入力させ、パーザで解析する。その結果

(6a) 正解なら、(4a) へ。

(6b) これでも誤っている場合、学習者の入力に最も近い正解とのずれについて、バグカテゴリに依存した媒介語テンプレートによるコメントを提示する。

3 誤り診断パーザ

本システムの誤り診断パーザは、穴埋め作文入力の解析に於いて、穴埋め部分に付けられた状況・機能インデックスを制約として利用して、診断を行ない、誤答入力に対して、最も近い正解を与えて、バグ仮説に基づいたコメントを提示できることを目指している。

実際、下位範疇的、状況機能意味的、位置的制約が十分に書き込まれた辞書と構文規則を利用したパーザを作り、学習者には、辞書が用意された自立語のリストを指定して作文させるとしても、なお、話者、聞き手、話題の間の関係や談話の流れに関する状況と、表現が果たすべき「機能」が指定されなければ、特に日本語のような状況依存度の高い言語に於いては正解と誤りを正確に仕分けるのは不可能である。

日本語の係り関係には、自立語へ係る句がその自立語の直前に並ぶという極めて単純な性質がある。この性質のために、日本語の解析は、スタックを用いた shift-reduce パーザ [2] によって行なうことによって極めて把握しやすいものになり、また学習者に対して、日本語教育の現場での説明に近いコメントを与える

ことができるものとなる。本システムでは、[3] を参考にして、このような特徴を利用した、次に述べるようなスタックパーザを作っている。

3.1 パーザの動作

まず、パーザの動作の概略を説明する。なお、本システムでは自立語を、リストから引かせるので、形態素解析は不要である。さて、語の辞書引きにより構文カテゴリ、下位範疇化拘束、位置拘束、意味拘束を得る。機能表現形式を構成する語の辞書には、この機能と状況制約も書き込まれるが、格助詞については、その機能がその係る用言に依存するので、表層のまま取り扱う。表 1 に「あげます」と「教えて」の辞書項目の例をあげる。但し、テンスについての記述は省略する。

自立語(用言あるいは体言)、その複合、あるいはそれに補助要素を伴うものは複合項と称する(例えば:「買ってあげた」)。一方、スタックには、項に係る接続要素が積まれる。そして、辞書引きによって、項が見出される度にスタック先頭から引き続いて、それに係ることのできる(一般に複数の)接続要素を積み下ろして句を作り、この句に、後続の接続辞を接続して新たな接続要素とする。その時、接続辞がなければ、句に空の接続辞を接続し、接続要素とする。こうして作られる接続要素をスタック先頭に積み、係りうるかどうかを調べる際には、構文的、前後関係の、ならびに状況・機能意味的拘束を利用する。引き続き接続要素を調べて、係れないものが現れたら、それ以上はスタックに残す。これによって交差係り受け禁止則を実現できる。

係り受け処理においては、補語(格助詞句)に対する用言の格支配の処理、用言修飾語/句、体言修飾語/句の助詞、意味素性、活用などの検査のほか、次の項目を、ほぼ実現している。

(1) 右外置検査: 例えば、「僕が仲間ちゃんに買ってあげたおもちゃ」では、用言項「買ってあげた」への係り受け検査に於いて、用言項が要求する「を」格が欠落していることを格情報[X, を, NIL]として記録した連体形の接続要素としてスタックに積み、その直後に体言項「おもちゃ」への係り受け検査に於いて、この格情報とXの意味素性を調べて、「おもちゃ」が右外置であることを確認する。

(2) 格の乗取り: 例えば、「私が田中さんに教えてもらう」では、主動詞「教える」の「が」格、「に」格にあった「田中さん」、「私」が補助動詞「もらう」

の意味表現の giver, givee スロットに入るので、「もらう」の「に」格「が」格の支配に変更されるというルールがある。これを「教えてもらう」という用言複合項で、予め格関係を修正した上で係り受け検査を行なうことによって実現している。

(3) 連体詞句の係り：例えば、「僕の母」では「の」の辞書の意味部のオプション [[be_kin,K, kinship,X, of,Y],.....] と前後主辞 [Y,X] 情報、そして、Y,X の意味素性を「僕」、「母」に合わせることで、適切な意味を決定している。

(4) 補語（格助詞句）の省略：話し手が関わる「が」格補語は省略して良い。依頼／命令の相手が聞き手であれば、聞き手が関わる「が」格は必ず省略されることなどは、インデックスの状況記述の locution 情報を利用して扱っている。

スタックパーザは、汎用のものとして作られるが、辞書情報と状況・機能インデックスを、定められた形式とプリミティブレパートリを利用して追加／変更することによって、現場の要請に合わせることが出来る。

スタックパーザでは、用言／体言に接続する句を接続要素 (Element) としてスタック (Stack) に積む。

◎ Stack ::= [Element | Stack]

Element ::= [Synstruct, Inflection, CaseConnectInfo, ModalityInfo, PrePost, Head, Semantic, SituationFocus]

但し、Synstruct: 統語構造。Inflection: 活用。CaseConnectInfo: 格または、接続情報。ModalityInfo: モダリティ情報。PrePost: 前後関係情報。Head: 主辞変数。Semantic: 意味表現。SituationFocus: 状況視点を、それぞれ表す。統語構造とは、句の構文情報である。また、活用は、用言類の活用情報である。格情報、接続情報は、右外置の為の欠落や、格助詞、接続助詞の使用に関する情報である。モダリティ情報は、話者の態度の情報のことである。前後関係情報は、連体詞句、連体接続句、連用接続句などにおいて、その句の主辞変数と、その後で、それが係るべき主辞変数をペアにしたもので、この句の意味表現に、両変数が含まれることによって、この後ろの接続部分との意味上の連絡をとるためのものである。主辞変数は、句における意味的な中心となる自立語に対する意味変数で、統語解析を通じて意味表現を接続していく役割を担う。状況視点は、句が成立するために成立すべき状況内での条件、例えば共感度条件を表す。

診断は、接続要素を作ってスタックに積む段階で

表 1: 辞書項目の例

「あげます」と「教えて」の辞書
<p>引数の順序は、先頭の品詞と最後の表層差分リスト以外は Stack の Element の構成素に対応している。</p> <p>dict('補助動詞',['補助動詞',X,'あげます'],'終止',[[W,'が'],[Y,'に']],['待遇','丁寧'],[['動詞'],'て形,Z],.), X, [[giving_benefit,X, giver,W, givee,Y, benefit,Z]], [empathy, >, W,Y], ['あげます' S],S)</p> <p>dict('動詞',['動詞',X,'教えて'],'て形',[[W,'が'],[Y,'に'],[Z,'を']],→X, [[teach,X, teacher,W, teachee,Y, what's_taught,Z]],→['教えて' S],S)</p> <p>(なお、W,Y,Z などの意味素性制約は、giving_benefit や teach に関する知識として別に書かれており、'教える'では[teach,teacher,teachee]の teacher,teachee が[giving_benefit,giver,givee]の giver,givee でもあることも知識として書かれている。)</p>

は、何等の判断も行わず、用言項あるいは体言項に対して係り得る接続要素をスタックから積み下ろすときに、表層上のバグカテゴリ毎に、バグ仮説のオプションについてその成立を調査する。パーザは、バグ仮説による修正で解析できれば、その修正情報をスタックに記録し、解析を続ける。もし、複数のオプションが成立するときは、そこまでのスタックを共有する複数の仮説としてのスタックを作って、並列に解析を進め、枝分かれで増大するスタックの内、矛盾が生じたものを捨てることになるが、この並列解析はまだ実装できていない。

3.2 誤り診断の為のバグ仮説

バグ仮説は下記のように分類される。

1) 活用の誤り

「て形」を要求する補助動詞が接続される用言の活用形の誤りなど。用言句につく接続助詞によって決まる用言接続、用言句の活用形の誤り。用言修飾用言句が連用形でない。体言修飾用言句が連体形でないなど

2) 機能語の欠落

状況・機能インデックスが要求する補助形式がない。用言項の要求する格の意味素性に合致する体言項に接続辞がついていない。用言修飾接続要素に、状況・機能インデックスが要求する接続辞がない。状況・機能インデックスが要求する意味での「の」がつくべき体言句についていない。

3) 機能語の誤り

補助形式が状況・機能インデックスに合っていない。体言句についている格助詞あるいは提題助詞が、用言項と状況機能インデックスの要求と合っていない。コンソアの指示詞が状況・機能インデックスの拘束に合っていない。用言修飾接続要素についている接続辞が、状況・機能インデックスに合わない。

4) 機能語の過剰

状況・機能インデックスが要求する以上の補助形式がある。用言修飾用言句に余分な接続辞がある。体言修飾用言句に「の」など余分な接続辞がある。

5) 句の欠落／誤り

状況・機能インデックスが要求する用言修飾句／体言修飾句がない／要求と合わない句がある。必須格の補語で省略不可のものが欠けている。体言項の意味素性が体言修飾用言句中の欠落格（右外置痕跡）の意味素性に合わない。

6) 句の過剰

状況・機能インデックスが要求する以上の用言修飾句／体言修飾句がある。体言項の主辞と同じものが体言修飾用言句中の補語基として存在する（右外置痕跡が埋められている）。

尚、自立語は与えられるのでその欠落や過剰はない。

3.3 誤り解析の例

一例として、状況・機能インデックス（の一部）が状況：[locution,l,locuter,y,apropos_of,g]

empathy: $y > w$

機能：[giving_benefit,g,giver,w,givee,y,benefit,t]

[teaching,t,teacher,w,teachee,y,what's_taught,j]

w: 田中さん, j: 日本語, であるとき, 入力文:

「田中さん、私、に、日本語、教える、あげます」に対する診断パーザの動作を述べる。但し表1のほか、利益授受表現「くれます」、「もらいます」の辞書には

「くれます」, [[w, が], [w, に]], [前にT主辞動詞「て形」], [giving_benefit, X, given, W, givee, Y, benefit, T], empathy: $Y > W$

「もらいます」, [[Y, が], [W, に]], [前にT主辞動詞「て形」], [giving_benefit, X, giver, W, givee, Y, benefit, T], empathy: $Y > W$

用言項「教えるあげます」が見出された段階で、スタックには「田中さん」、「私に」、「日本語」の三つの接続要素が積まれている。「教えるあげます」は、意味表現はインデックスと合うが、empathy条件が、「あげます」の辞書からは $w > y$ であり、インデックスからは $y > w$ なので、合わない。そこで、「機能語の誤り」のバグ仮説を適用して、意味とempathyの合う機能語として「くれます」、「もらいます」を見出す。しかし、「教えるくれます／もらいます」は、その前にtを主辞とする動詞に「て形」を要請することに反するので、「活用の誤り」のバグ仮説を適用して、「教えてくれます／もらいます」への修正コメントを記録する。その格支配は、「教えてくれます」では, [[w, が], [y, に], [t, を]], 「教えてもらいます」では, 乗っ取りがあつて, [[y, が], [w, に], [t, を]] なので、前者に対しては「機能語の欠落」のバグ仮説により、「田中さん」「日本語」を「田中さんが」、「日本語を」に、後者に対しては「機能語の欠落と誤り」のバグ仮説により、「田中さん」、「私に」、「日本語」を、「田中さんに」、「私が」、「日本語を」へ修正するコメントを記録する。そこで、修正量が少なし「田中さんが、私に、日本語を、教えてくれます」を正解の第1候補とする。

4 おわりに

今後の課題としては、バグ仮説を、実際の現場でとられた事例をもとに詳細化することにより、確実な信頼のおける修正コメントを作れるようにしてゆくこと。バグ仮説レベルでの修正の統計に基づいた学習者モデル構成法が挙げられる。

参考文献

- [1] 劉 軼, 榎本 圭孝, 加藤 伸隆, 馬目 知徳, 伊丹 誠, 伊藤 紘二: “状況と機能に応じた日本語表現の学習を支援するシステム,” 電子情報通信学会論文誌, Vol.J80-D-II, No4 (1997.4)
- [2] 田中 穂積: 自然言語解析の基礎, 産業図書, 1989.
- [3] 黒橋 禎夫, 長尾 真: 並列構造の検出に基づく長い日本語文の構文解析, 自然言語処理, Vol.1, No.1, pp.35-57, 1994.