九州大学学術情報リポジトリ
Kyushu University Institutional Repository

動的な教材開発を目指して(6)： ドイツ語の叙述的形容詞と副詞、および語順を解析する

栗山, 暢
九州大学大学院言語文化研究院言語環境学部門 ： 准教授 ： 言語教育学

https://doi.org/10.15017/21798

出版情報：言語文化論究. 28, pp.61-72, 2012-03-02. 九州大学大学院言語文化研究院
バージョン：
権利関係：

動的な教材開発を目指して (6)

――ドイツ語の叙述的形容詞と副詞、および語順を解析する――

栗　山　　暢

九州大学大学院言語文化研究院　言語文化論究　第28号　平成24年2月発行　抜刷

Faculty of Languages and Cultures, Kyushu University
Motooka, Fukuoka, Japan

STUDIES IN LANGUAGES AND CULTURES, No.28, February 2012

動的な教材開発を目指して（6）

― ドイツ語の叙述的形容詞と副詞、および語順を解析する ―

栗　山　　暢

1．目標と考え方

コンピュータを使った教育に普通に見られる、「選択式」あるいは「完全一致式」の問題形式に変わるものとして、筆者は、学習者の書いたドイツ語の作文をある程度知的に添削するコンピュータプログラムの開発を目指している。これまで解析してきたのは、名詞句（前置詞に支配されるものを含む）と動詞句のみによって構成される単文であったが、本稿では、これに叙述的形容詞と副詞の解析を加え、さらに単文全体の語順を解析することを目指す。

1．1　叙述的形容詞と副詞の解析

"der Mann ist stark" における形容詞 "stark"、"der Mann kommt morgen" における副詞 "morgen" は、ともに、独立した成分として文に直接かかわっているように見えるという意味で、コンピュータで処理する際にはまったく等価であると言ってよい。あるいは、"der Zug ist schnell"、"der Zug fährt schnell" におけるふたつの "schnell" も、コンピュータ処理上では、それぞれの品詞がなにであるかということに頭を悩ませるまでもなく、まったく等価に扱うことができるだろう。叙述的形容詞が文中に現れるのは通常たかだか１回であるのに対し副詞は何度でも出現し得るけれど、その事実もそのふたつを同等に扱うことをなんらさまたげない。本システムでは、当面（そうすることによる不都合が見つかるまで）、叙述的形容詞と副詞とを、解析の過程では区別しない。

もちろん、"der Mann ist stark" という答えが期待されている時の "der Mann ist" という解答（"stark" を使用しないという誤ち）と、"der Mann kommt heute nicht" という答えが期待されている時の "der Mann kommt nicht" という解答（"heute" を使用しないという誤ち）とでは、誤ちの種類ないし重要性が異っているとしなければならないだろう。文に不可欠の成分と、自由に付加される成分とは、区別する必要がある。これは、解析の問題ではなく、解析結果の解釈の問題である。

解析にあたってほかの形容詞・副詞と区別しなければならないものは、それ自身がほかの形容詞・副詞を修飾し得る副詞である。"der Zug fährt sehr schnell" という文における "sehr" は "schnell" を修飾しているとしか考えられず、"sehr" と "schnell" が同等の立場として独立して文にかかわっていると解析してはならない。自明のことのようだが、"der Zug fährt schnell sehr"（前から修飾すべき副詞が後ろから修飾している誤ち）のような文を書く初学者もめずらしくはないのだから、このことにもきちんと対応しなければならない。

副詞を扱うことによって、"nicht" を含む否定文の解析が可能になるから、否定冠詞 "kein" を使うべきところで "nicht" を、あるいは反対に "nicht" を使うべきところで "kein" を使っているような誤りも捕捉したい。

１．２　語順の解析

ある文において唯一の正しい語順というものがドイツ語の場合つねに考えられるわけではないだろうが、もっともありそうな、もっとも自然な語順を想定することはいつも可能である。したがって、解答者の語順が出題者の期待に沿うような語順になっていない場合、期待する語順に向かってゆるやかに誘導するようなメッセージを与えることが望ましい。明らかに誤りである語順と、明らかに誤っているわけではないが望ましいものでもない語順とは区別して、それぞれに応じた反応を返せるようにしたい。

明らかに誤りである語順としては、定形の位置に関する誤りのほか、いわゆる枠構造を形成する結びつきに違反する次のみっつの場合を特別にチェックしたい。

1）動詞と自立性のない目的語の結びつき

2）動詞と方向規定句の結びつき

3）コピュラ動詞と述語句の結びつき

すなわち、"er ist krank heute" や "er geht in die Stadt morgen" などという解答に対し、一般論として語順が間違っていると指摘するのではなく、明示的にそれぞれ上の 3)、2）の規則に違反していることを告げるメッセージを出すこととしたい。

ただし、実質的に同じ文だが、"er ist krank nicht" や "er geht in die Stadt nicht" に対しては、枠構造の誤りではなく、"nicht" の位置の誤りと指摘した方がよいかも知れない。やることは同じだが、初学者のいだく実際的な関心は、文全体の構造にかかわるものであるよりは、"nicht の位置 " にかかわるものであることが多いと思われるからである。

"nicht" の位置についての誤りに関しては、このような場合だけでなく、常に、それが修飾すべき単語の直前に位置しているかどうかもチェックしたい。"der Mann kommt morgen nicht" が期待されている時の "der Mann kommt nicht morgen" という解答は、単に一般的な語順の誤り以上に重要な（しかも明白な）誤りを含んでいると考えられるからである。

上のみっつの規則が、主文の場合には実質的には文の一番最後に来なければならない要素に関するものであるのに対し、文頭に来なければならない要素として、疑問文における動詞の定形、疑問詞、文のテーマを表す句を設定したい。

また、2 格（属格）名詞が後ろから前の名詞を修飾する場合の語順についての解析もおこなう。初学者には不要だと考えていいいわゆるザクセン 2 格は認めないことにし、すると、"des Kindes der Vater" などという誤りも語順についてのみ考えればよいことになる。しかし、これも、"der Mutter der Tochter" という答えが想定されている時に "der Mutter der Mutter"（Tochter と書くべきところを Mutter と書いた誤り）などという解答が与えられた時には、やや複雑なことになる。栗山(2011b) でおこなった名詞句の（問題の情報と解答の名詞句の組み合わせを決定する）解析方法では、このふたつの "Mutter" の区別をつけることはできず、結果、「2 番目の "Mutter" は "Tochter" の誤りである」という正しい反応を確実に返すことができない。2 格を含む語順を考慮することによって、この点を改善したい。

これまでは FiniteComparer クラスにおいて行っていた定形の位置に関する解析や、Parser クラスが行っていた最適の構造の選択も、副詞や語順の解析と密接にかかわってくることになるだろう。"so viele Leute kamen gestern" の定形の位置が正しいかどうかは "so" の解釈にかかわっているし、反対に、"so" を解釈する時には定形の位置を考慮した方がよい。"der Mann ist stark der Mann" などという文において、ふたつの "Mann" のうちどちらが文に不要な成分であるかを指摘するには、語順の情報によるしかない。名詞句、動詞句、定形、今回とりあげる副詞・形容詞（句）そして語順

の解析を単に連続して実行するのではなく、それらから得られるデータを総合することによって、学習者の意図をもっともよく反映した反応を返すシステムをめざしたい。

２．実装

使用した言語は Ruby（version 1.9.2）。今回追加したファイルは以下の通り。ひとつのファイルがひとつの（Ruby の）クラスに対応している。

```
lib/word/adverb.rb
lib/dictionary/make_adv_data.rb
lib/question_info/adj_adv_info.rb
lib/question_info/adj_adv_info/adj_as_adv_info.rb
lib/question_info/adj_adv_info/adverb_info.rb
lib/question_info/adj_adv_info/predicative_adj_info.rb
lib/question_info/attributive_adj_info.rb
lib/parser/adj_adv_phrase.rb
lib/parser/adj_adv_comparer.rb
lib/parser/sequence_comparer.rb
lib/parser/sequence_comparer/structure_for_sequence.rb
lib/fallacy_weight_decider.rb
```

２．１　叙述的形容詞・副詞の解析

単語としての形容詞の情報を保持するクラスは、限定的形容詞と同じ、Adjective クラスである。副詞の情報を保持するクラスとして Adverb を作った。Adverb クラスのインスタンス（個々の副詞）は、原形や級といった情報のほかに、1）ほかの副詞や形容詞を修飾し得るか、2）ほかのどの副詞に修飾され得るか、の情報を持つ。"freilich" という副詞はほかの副詞を修飾せず、ほかの副詞に修飾もされないが、"sehr" という副詞はほかの副詞や形容詞を修飾し、かつみずからも "zu" や "so" などに修飾される、といった情報である。副詞は数も少なく、その有無が文の構造に重大な影響を与えることも少ないので、名詞・動詞・形容詞とは異なり、すべての副詞を初めから辞書に登録しておくことにした（すなわち、問題の情報によって与えられていない副詞も受けつけることとした）。

解答の辞書ひきをして得られた単語列の可能な品詞の組み合わせをすべて生成する Structures クラスは、ある品詞が形容詞の場合、従前の :adj という（品詞の組み合わせの文法的整合性をチェックする Structure クラスための）ラベルづけをするだけでなく、新たに、:bare_adj というラベルづけをすることになる（:bare_adj は叙述的形容詞を表し、従来の :adj が限定的形容詞を表す）。副詞に関しては、文の独立した成分として働く、ないし名詞句の外の形容詞・副詞を修飾とする場合の :adv と、名詞句の中に現れる形容詞を修飾する場合の :adv_in_np というふたつのラベルづけをする。たとえば、"sehr viele Leute kamen" という単語列に対し、

1）[:adv :adj :noun: :verb]
2）[:adv_in_np :adj :noun: :verb]

3）[:adv :bare_adj :noun :verb]
4）[:adv_in_np :bare_adj :noun :verb]

という計よっつの組み合わせが生成されることになる（Structure クラスが文法的に正しいと認めないのはこのうち、:adv_in_np と次に出現する :noun の間に :bare_adj が出現する 4）だけである。ちなみに、Structure クラスが生成する文の構造は、1）[:ADV, :NP,:VERB]、2）[:NP, :VERB]、3）[:ADV, :BARE_ADJ, :NP, :VERB] である）。

この品詞名のうち、:bare_adj と :adv を解析するものが AdjAdvComparer クラスであり、:adv_in_np は従来の NpComparer クラスで解析する。後者は前者を単純にしたものであるに過ぎないので、本稿では前者についてのみ解説する。

２．１．１　AdjAdvComparer

AdjAdvComparer は、まず、解答に現れる副詞が、直後に位置する副詞あるいは形容詞（以下、誤解の生じないところでは叙述的形容詞を単に形容詞と呼ぶことにする）を修飾する可能性があるかないか判定する。以下のよっつのいずれかにあてはまる時には次の単語を修飾していないと判断する。

---

1) 次の単語が副詞・形容詞でない
2) 次の単語が当該の副詞によって修飾されることのない単語である
3) 次の単語が問題の情報によって与えられている副詞・形容詞でない（本稿においては、ほとんどの場合、問題の情報によって与えられている、とは、文の要素として独立したものとして要求されているということを意味している。ほかの副詞・形容詞を修飾する副詞は文に不可欠なものではないので、修飾される単語に付随する情報と考える）
4) 次の単語を修飾すべき副詞がこの副詞ではなく、かつ、この副詞がほかの副詞を修飾する性質を持たない

修飾している可能性がある時にはほかのことを一切考慮せずに、その副詞は次の副詞・形容詞を修飾していると判断する。"viele Leute freuen sich sehr" という答えが求められている問題に対して "sehr viele Leute freuen sich" という解答が入力された時、問題の情報を持っているからといって "sehr" が "freuen" を修飾していると判断してその結果定形の位置の誤りを指摘するのはかなり不自然なことのように思われるからである。解答者が "sehr" が何を修飾しているつもりであるかはいずれにしても不明であるので、ドイツ語解釈としてより自然に思われる方を選択するべきだろう。

後ろの単語を修飾していると判断した場合は、修飾される側の単語が前の副詞に修飾されることが求められているか（誤った副詞によって修飾されていないか、そもそも修飾されることが求められていないのに修飾されているのではないか、などを含む）、級は正しいかなどを判断する。

後ろの単語を修飾していないと判断した場合は、当該の単語が問題の情報によって与えられているものかどうか判断する。そうであれば、その単語自身が別の副詞によって正しく修飾されているか（正しく修飾されていないか）をチェックし、級が正しいかも判断する。そうでなければ（この単語はすでに後ろの単語を修飾しないことがわかっているので）誤って前の副詞・形容詞を後ろから修飾している可能性をチェックする。誤って前の副詞・形容詞を後ろから修飾すると判断するのは、前の副詞・形容詞がさらにその前から修飾されておらず、かつ、当該の副詞によって修飾されるこ

とが求められている場合のみとした。"er ist stark" が正解であるような時の（"stark" が "sehr" に修飾されることが求められいない場合）"er ist stark sehr" という解答には「"sehr" は修飾する単語の前に置く」というコメントを返した方がよさそうにも思えるけれど、より保守的な（学習者には情報量の少ない）「"sehr" は不要である」という反応を返す。"sehr" をどのようなつもりで使っているかがやはり判然としないからである。

２．２　語順の解析

２．２．１　２格名詞の語順の解析

２格名詞の語順を解析するために、当該の２格名詞をともなうべき名詞の情報にその旨記載することとした。"Vater:noun:der:s:sub:gen::Kind:noun:der:s:2" の "gen" がそれである。これにより、NounInfo で、当該の２格名詞の @modifier 属性を true に、それをともなう名詞の @has_genitive 属性を true にする。２格名詞の語順にかかわるのは名詞のみなので、NpComparer がその解析を行う。

NpComparer では、栗山（2010b）で解説したこれまでの名詞句解析をすべての名詞について終えた後、@has_genitive 属性が true の名詞それぞれにつき、その前後に名詞があるかないか調べ、両方に名詞がない時には NP_NO_GENITIVE_ERROR を与える。後ろに名詞があり、それが正しい２格名詞である時には正しく接続されているものとし後ろの（２格の）名詞に :rightly_modifies 属性を与える（名詞の種類が正しいが２格でない時には２格であるべき名詞に NP_CASE_ERROR が与えられているのでなにもしない）。後ろの名詞に :rightly_modifies 属性が与えられなかった時には前の名詞について同じことをするが、格および名詞（の原形）が合致した時には２格の名詞に NP_GENITIVE_MODIFIES_FROM_FORWARD_ERROR を与える。その後今度は２格であるべき名詞すべてについて :rightly_modifies 属性が与えられているか調べ、それが与えられておらずまた NP_GENITIVE_MODIFIES_FROM_FORWARD_ERROR も与えられていない時には NP_GENITIVE_WRONG_POSITION_ERROR を与える。

栗山（2011b）で解説した、同一の名詞が複数回出現する場合の問題の情報と解答の名詞との対応づけの方法は、最終的にはなるべく曖昧さを残さないようにするものであり、最後まで組み合わせを決定できない時には恣意的に決定していた。今回語順の解析をすることにより、それによって得られた情報によってよりよい組み合わせを決定できることが期待されるようになるのだから、NpComparer で無理に組み合わせを決定することはむしろ好ましくない。そこで、今回加わった２格の語順に関する誤りをも含め、あらゆる組み合わせをそこに含まれる誤りの総数で比較し、誤りの数のもっとも少ない組み合わせを、それが複数であっても、候補として残し、最終的な決定は SequenceComparer にゆだねることとした。ただし、栗山（2011b）の方法もその時点では十分な効果をあげていたのだから、誤りの数で候補がひとつにしぼれなかった場合には、主語に関する情報（主語であるべき名詞が１格（主格）でない誤りを重視する）、形容詞に関する情報（使われるべき形容詞が使われていない誤りを重視する）、前置詞に関する情報（3（属格）、4 格（対格）支配の前置詞にあって名詞が３ないし４格であるが正しい格でない誤りを軽視する）、格の誤りに関する情報（格の誤りを重視する）をこの順に考慮して候補をしぼっていくことにした。

２．２．２　文の語順と名詞および副詞・形容詞の組み合わせの決定

そのほかの語順の解析は主に SequenceComparer クラスが行う。

これまでは問題の情報の個々の単語の情報の配列は任意であったが、今後はもっとも望ましい語順の順に単語の情報を並べることとする。問題の情報に与えられている個々の要素（単語）を順に 0 から数えることにすると、n 個の要素がある時に、解答の句が 0 1 2 … n-1 と並んでいる時に正しい語順であると判断する。

たとえば、"Anfang:noun:prep=an::sein:verb::Wort:noun" という問題の時（ここでは必要な情報のみ示した。以下同じ）、この場合、問題の情報と解答の単語との組み合わせは一意に決まるから（"Anfang", "Wort" とも問題、解答の両者の中に一度づつしか出現しないため）、"am Anfang war das Wort"、"das Wort war am Anfang" というふたつの解答における問題の要素の並び方は、

"am Anfang war das Wort" 0 1 2

"das Wort war am Anfang" 2 0 1

となる。前者が解答として与えられた時には何も指摘せず、後者が与えられた時には語順の誤り（SENT_SEQUENCE_ERROR）を指摘する。

名詞と副詞・形容詞の問題と解答の組み合わせが未決定の場合、それを最終的に決定するのにも語順の解析結果を使用する。すなわち、可能な組み合わせすべての組み合わせについて、その並び順と問題の情報によって指定されている並び順（0 1 2 … n-1）とがもっとも似ている組み合わせを最終的な組み合わせとする。並び順の類似度を判定するには編集距離を使う。解答の句の並び順を文字列として（たとえば "210" のように）正しい並び順の文字列（"012"）と比較して、その間の編集距離のもっとも短いものを選択するのである（もちろん、10 を越える句を越える時にも処理できるよう、実際には "0" を "a"、"1" を "b"、"10" を "j" などとおきかえる）。

上の問題に、"am Anfang das Wort war das Wort" という解答が与えられた場合（余分な単語 "Wort" が誤って加わっている）、2 番の情報（"Wort" に関する）を、解答の、1 番、3 番のどちらの "Wort" と組み合わせるべきか、ここで決定する。1 番と組み合わせると 4 番の "Wort" が不必要な単語だということになり、4 番と組み合わせると 1 番が不要な単語だということになる。両者における並び順はそれぞれ "021"、"012" であるから、正解（"012"）との編集距離のもっとも短い（この場合 0 である）後者を選択し、最初の "Wort" に対して「不要である」というメッセージを返す。

ただし、編集距離だけを規準にすると問題が生じる場合もある。"das Wort war am Anfang das Wort" という解答では、ふたつの "Wort" のどちらを選択しても（最初のを選択すると "201"、二番目のを選択すると "102"）、得られる編集距離は同一である。二番目のを選択したとすると（最初の "Wort" を不要と判断する）、最初の "Wort" に関して「不要である」というメッセージを出す一方、定形の位置に関する誤りも指摘することになる（最初の "Wort" を無視するため）。このような望ましくない事態は、定形の位置が正しい候補があればそれだけについて編集距離を比較することによって避けることができる。結果、最後まで組み合わせを決定できないのは、"am Anfang war das Wort das Wort" のように、実際にどちらを選んでもさしつかえない場合だけであることが期待される。

このようにして最終的に決定された名詞と副詞の問題情報との組み合わせについて、SequenceComparer はその組み合わせにおける誤りの情報を、それぞれ NpComparer、AdjAdvComparer にセットする。

なお、SENT_SEQUENCE_ERROR は情報量が少ないため、文頭・文末に置かれるべき要素や否定の副詞の位置に関する誤りなどがない時にのみ指摘することにした。

２．２．３　文頭・文末に置かれるべき要素

上の問題に、"Anfang:noun:prep=an:top=th::sein:verb::Wort:noun" と、"top=th" という情報を加えることによって、"Anfang" が文頭に置かれていない場合には、SENT_SEQUENCE_ERROR でなく、SENT_TOP_WORD_POSITION_ERROR を与えるようにした。栗山（2011b）で述べた通り、すべてのエラーメッセージは ErrorMessage クラスのサブクラスのインスタンスとして、個別の情報とともに生成される。こうすることによって、このような場合、文のテーマ、疑問文の定形、疑問詞は文頭に置かなけらばならないと具体的に指摘できるようにした。

文末に置かれるべき要素についても、同様に、"last=cop" などと問題に記述することによって、語順の誤りを理由とともに示せるようにした。文末の場合、"er ist stark sehr"（"stark" が "sehr" に修飾されることが問題の情報によって求められているとする）などのような時には、sehr の位置の誤りをのみ指摘する。nicht の位置との関係についても同様に扱うことにした。

２．２．４　定形の位置

"sehr stark ist er" という文を、Structure クラスが文法的に適格と認めた構造のうちのひとつ、[:ADV, :BARE_ADJ, :VERB, :NP] という構造で解析する時、"sehr" が "stark" を修飾しているのか文（動詞）を修飾しているかは、AdjAdvComparer によってすでに決定されている（この場合は前者）。Strucutere クラスの文の構造の解析にはこのように副詞の修飾関係の情報は含まれていないので、SequenceComparer クラスは、AdjAdvComparer クラスによって得られた情報を統合して、新たな構造を決定する。この場合は [:BARE_ADJ, :VERB,:NP] という構造が得られることになり、この副詞の修飾関係の曖昧性を除去した構造に基づいて定形の位置を判断する。以前 FiniteComparer クラスで判断していた時には３番目に位置するとされていたこの文における定形は、ここにおいて２番目に位置すると（正しく）解釈されることになる。

２．２．５　否定の副詞の語順

否定の副詞に関しては、そのほかの副詞とは別に、独自に解析を加える。否定の副詞は否定される単語の直前に置かれるという規準にもとづき、解答の問題情報列において、否定の副詞の問題の情報における位置が i ならその後に i ＋ 1 が来ていない場合に（i が一番大きい数字の時は解答の問題情報列の最後に来ていない場合に）、ADV_NEGATIVE_ADV_POSITION_ERROR を与える。

"Vater:noun:1::geben:verb::Kind:noun:3::nicht:adv::Buch:4" という問題に、"der Vater gibt dem Kind das Buch nicht" という答えが与えられた場合、解答の問題情報列は ０１２４３ となり、"nicht" の位置である３の後に４が続いていないので、この誤りが検出される。

文末に来なければならない要素があり否定の副詞がそれを修飾しなければならない場合に、解答において否定の副詞が文末に、文末になければならない要素がその直前に置かれた場合には、特に ADV_NEGATIVE_ADV_AFTER_LAST_WORD_ERROR を与えることにした。これは文末に来なければならない要素に対する SENT_LAST_WORD_POSITION_ERROR と否定の副詞に対する ADV_NEGATIVE_ADV_POSIITON_ERROR とを一度に指摘していることになる。

２．３　文の構造の決定

SequenceComparer までは、ひとつの問題と解答について考えらるすべての（Structure クラスが適格だと判断したすべての）構造について解析をしている。2.1 であげた、"sehr viele Leute kamen" という解答については、副詞の曖昧性を除去した構造で示せば、

1）[:ADV, :NP, :VERB]
2）[:NP, :VERB]
3）[:BARE_ADJ, :NP, :VERB]

となる。

このうち最適の構造を選択するのは Parser クラスである。栗山（2011b）、栗山（2011c）では、動詞句の文法的適格性、前置詞と前綴りの競合、名詞句の誤りなどに関し、それぞれ独立に、順番にチェックして、最初に候補の数が 1 となった時にそれを最終的な構造として選択した。今回それに副詞・形容詞の解析、語順の解析結果を加えて同じように判断しても、しかし、最適の構造が選択される保証はない。今度はたがいの解析が完全に独立しているとは考えられないからである。

そこで今回は、個別の解析の積み重ねは放棄して、ある構造が全体として持つ誤りの量にもとづいて最適な構造を選択することを試みた。まず、文の構造に重大な意味を持つと考えられ、また他の解析から独立していると思われる動詞句の文法的適格性と、必ずしも誤りにもとづいて構造を選択しているわけではない前置詞と前綴りの競合の解析を従来どおりに行った後、誤りの量にもとづいた選択をする。すべての誤りが構造の選択に同様に貢献するとは考えられないから、誤りの種類別に重みづけをすることにした。少数の、複数の構造を持った問題と解答を用意して、どのような重みづけをすれば期待される構造が選択されるか、実際に解析してみた。最初に用意した問題と解答、期待される構造は次の通りである。

"Mann:noun:_:p:sub:viel（sehr）::bringen:verb:ind_pres:act::morgen:adv::Buch:noun:der:s:4::nicht:adv"
"sehr viele Männer bringen morgen das Buch nicht" [:NP, :VERB, :ADV, :NP, :ADV]
"morgen sehr viele Männer bringen das Buch nicht" [:ADV, :NP, :VERB, :NP, :ADV]
"morgen sehr viel Männer bringen das Buch nicht" [:ADV, :NP, :VERB, :NP, :ADV]
"morgen viele Männer bringen das Buch nicht" [:ADV, :NP, :VERB, :NP, :ADV]
"morgen viel Männer bringen das Buch nicht" [:ADV, :NP, :VERB, :NP,:ADV]

"Mann:noun:_:p:sub:viel::bringen:verb:ind_pres:act::morgen:adv::Buch:noun:der:s:4::sehr:adv"
"viele Männer bringen morgen das Buch sehr" [:NP, :VERB, :ADV, :NP, :ADV]
"sehr viele Männer bringen morgen das Buch" [:NP, :VERB, :ADV, :NP]

この 7 例の解答において、構造の解析の解析に関与することになる誤りは、次の 8 個であった。

ADJ_ADV_NO_ADV_ERROR,
VP_FINITE_POSITION_ERROR,
SENT_EXTRA_ADJ_ADV_ERROR,
NP_UNUSED_ADJ_ERROR,

ADJ_NO_TAIL_ERROR,
ADV_WITHOUT_MODIFIED_WORD_ERROR,
ADJ_ADV_EXTRA_ADV_ERROR,
SENT_UNUSED_WORD_ERROR,

重みづけを行わない（すべての誤りの重さを 1 とする）場合、正しい構造の選択はできなかった。最大の重みを 2 とした場合(誤りの重さを 1 または 2 とする。誤りの重さの組み合わせは 2^8 通り)も、正しい構造の選択はできなかった。最大の重さを 3 とした時に、全部で 6561（3^8）通りの重さの組み合わせの中で 92 通りの組み合わせにおいて、期待される構造が選択された。

２．４　nicht と kein

nicht で否定するべきところを kein を用いて否定している、およびその反対の誤りの捕捉は、すべてのエラーメッセージのインスタンスを生成する MessageMaker において行う。SENT_EXTRA_ADV_ERROR の実体が否定の副詞であり、一方名詞句のどれかが、使われるべき冠詞として "kein" を持っている NP_ARTICLE_KIND_ERROR を持つ時、当該の単語からそれぞれの誤りを取り除き、文全体に、SENT_NICHT_TO_KEIN_ERROR を与える。反対の場合も同様である。

３．結果

以上を含め、全体を実装するのに約 18000 行を要した。

実際の解析例をいくつか紹介する。解析はおおむね期待どおりになっているけれど、若干の問題点をその都度指摘する。"=>" の左側が解答、右側が捕捉された誤りで、"{ }" の中の "=>" の前の数字は解答の中の単語の位置（0 から始まる）を示している。:general は文全体にかかわる誤り、:unused は使用されなかった単語を示すものである。問題の情報をすべて表示するのは繁雑なので、正解（例）をその代わりとし、必要な時にのみその他の条件を示した。

正解 "viele Männer denken so"
1）　"so viele Männer denken" => {0=>["ADJ_ADV_EXTRA_ADV_ERROR（so）"], :unused =>["SENT_UNUSED_WORD_ERROR（so）"]},

正解 "so viele Männer kommen"
2）　"viele Männer kommen" => {0=>["ADJ_ADV_NO_ADV_ERROR（so）"]}
3）　"zu viele Männer kommen" => {0=>["ADJ_ADV_WRONG_ADV_ERROR（[so, zu]）"]}
4）　"viele Männer kommen so" => {0=>["ADJ_ADV_NO_ADV_ERROR（so）"],
　　3=>["SENT_EXTRA_ADJ_ADV_ERROR（so）"]}

正解 "der Mann singt sehr schön"
5）　"der Mann singt schön sehr" => {4=>["ADV_MODIFIES_FROM_BACKWARD_ERROR"]}
6）　"der Mann singt schön so" => {3=>["ADJ_ADV_NO_ADV_ERROR（sehr）"],
　　4=>["SENT_EXTRA_ADJ_ADV_ERROR（so）"]}

(3) のように、誤りの名前の後ろの括弧の中に "[..]" とあるものは、その中の前者が正しいもの（この場合は単語）、後者が誤ったものを示している。）

副詞についての解析は、意図どおりになっている。1）だけ見ると、かたや "so" が足りない、かたや "so" が余分であると指摘しているので、ひょっとすると "so" の位置の誤りを指摘するのがよいようにも思えるが、不足しているものと余分なものの関係において反対の構造になっている 4）を見るとはたして本当にそうするべきか疑問にも思える。この場合は文の意味がかかわっているようでもあり、それは本稿の目的をはるかに越えるものだから、解答者に与える情報量としては十分に多いとは言えないかも知れないが、現状で満足したい。このような場合には解答者の意図が不明だから、情報量は多すぎてはいけないと考えてもいいかも知れない。

正解 "der Mann der Frau ist so stark"
7） "der Frau der Mann ist so stark" => {1=>["NP_GENITIVE_MODIFIES_FROM_FORWARD_ERROR"]}
8） "der Mann so der Frau ist so stark" => {1=>["NP_NO_GENITIVE_ERROR"],
2=> ["SENT_EXRA_ADV_ERROR（so）"],
4=>["NP_GENITIVE_WRONG_POSITION_ERROR"]}

正解 "er dankt der starken Frau der Frau"（奇妙な文だが意味については考えないでもらいたい）
9） "er dankt der Frau der starken Frau" => {3=>["NP_GENITIVE_MODIFIES_FROM_FORWARD_ERROR"]}
10） "er dankt der Frau der Frau", {3=>["NP_UNUSED_ADJ_ERROR（[stark]）"]}

---

2 格の語順についての解析である。9）では、形容詞がついている名詞を 3 格、そうでない方を 2 格と解釈するのでこのような結果になる。10）は語順をたよりに、前の名詞を 3 格、後ろの名詞を 2 格と解釈し、3 格の方に形容詞の不足を指摘している。

8）は、2 格の位置が正しくないと指摘しているのだから、修飾される側の NP_NO_GENITIVE_ERROR は削除する方がいいのかも知れない。

正解 "der Mann ist heute sehr krank"（"krank" が文末に来るべきことが指定されている）
11） "der Mann ist sehr krank heute" => {3=>["SENT_LAST_WORD_POSITION_ERROR（:cop）"]}
12） "der Mann ist heute krank sehr" => {5=>["ADV_MODIFIES_FROM_BACKWARD_ERROR"]}

正解 "der Mann ist nicht krank"（"stark" が文末に来るべきことが指定されている）
13） "der Mann ist krank nicht" => {5=>["ADV_NEGATIVE_ADV_AFTER_LAST_WORD_ERROR"]}

正解 "morgen gibt der Mann dem Kind ein Buch"（"morgen" が文頭に来るべきことが指定されている）
14） "der Mann gibt dem Kind morgen ein Buch" => {5=>["SENT_TOP_WORD_POSITIOLN_ERROR(:th)"]}

正解 "morgen gibt der Mann dem Kind ein Buch"（語順に関する情報は問題の情報における単語の配列だ

けである）
15）"der Mann gibt dem Kind morgen ein Buch" => {:general=>["SENT_SEQUENCE_ERROR"]}

副詞の位置、文頭や文末に来るべき要素、否定の副詞それぞれについて、意図どおりの解析が行われている。
"nicht" と "kein" についても、次に見るとおり、解析は成功する。

正解 "der Mann hat kein Buch"
16） "der Mann hat das Buch nicht" => {:general => ["SENT_NICHT_TO_KEIN_ERROR"]}
17） "der Mann hat kein Buch nicht" 0> {5=>["SENT_EXTRA_ADJ_ADV_ERROR（nicht）"]}

正解 "der Mann hat das Buch nicht"
18） "der Mann hat kein Buch" => {:general=>["SENT_KEIN_TO_NICHT_ERROR"]}
19） "der Mann hat kein Buch nicht" => {4=>["NP_ARTICLE_KIND_ERROR（[der, kein]）"]}

以上、本稿で目的としたことはおおむね達成されたと言うことができるだろう。今後問題になりそうなのが、Parser における構造の選択の部分である。上に示した解析例を含め百の単位でさまざまな文を解析したところでは、2.3 で示した 8 個の誤りの最大 3 の重みづけで構造の選択に成功しているが、今後、予想もつかないような解答が入力された時に必ず成功するという保証はない。
だが、選択の失敗は候補をひとつにしぼれないという形で現れるから、それを捕捉するのは簡単なことである。そのような事態が発生した時には、当該の問題と解答の組み合わせについて、

1） すでに得られている 92 のそれまで機能していた重みづけの組み合わせのうち、未使用のものを試す
2） 当該の問題と解答の組み合わせで新たに関与することがわかった誤りがあればそれを含めた新たな重みづけの組み合わせを試す
3） 重みづけの最大値を増やすの順に検証すること

によって、新たに正しく機能する重みづけの組み合わせを発見することができると期待される。

参考文献

栗山　暢（2010a）. 動的な教材開発を目指して（1）-- ドイツ語の冠詞と名詞を解析する .『言語文化論究』, 25,45-52.
栗山　暢（2010b）. 動的な教材開発を目指して（2）-- ドイツ語の名詞句を解析する .『ドイツ語情報処理研究』, 20, 25-35.
栗山　暢（2011a）. 動的な教材開発を目指して（3）-- ドイツ語の動詞句を解析する .『言語文化論究』, 26, 49-57.
栗山　暢（2011b）. 動的な教材開発を目指して（4）-- ドイツ語の単純な単文を解析する .『ドイツ語情報処理研究』, 21, 37-48.
栗山　暢（2011c）. 動的な教材開発を目指して（5）-- ドイツ語の前置詞と人称代名詞を解析する .『言語文化論究』, 27.（印刷中）

清野智昭（2008）.『中級ドイツ語のしくみ』. 東京 : 白水社 .