

英文の誤りを検出するシステム ASPEC-I†

河合 敦夫^{††} 杉原 厚吉^{††} 杉江 昇^{††}

本論文では、主として日本人が書く英作文の誤り（統語的および単語綴りの誤り）をみつけ、エラーメッセージを表示するシステム、ASPEC-I (an Advisory System for Polished English Composition I) について述べる。ASPEC-I では、補強文脈自由文法をゆるめることにより、統語的誤りをみつけている。科学技術論文を対象として、ASPEC-I の文法の有効性を調べたところ、英米人の書いた英文では 65% が構文解析可能であったが、日本人の書いた英文にたいしては、83% が可能であった。また、ASPEC-I を使って、日本人の書いた、添削を行っていない英文にたいして、誤り検出の実験を行った。その結果、意味や文体上の誤りも含む誤り全体に対しては、37% を検出したが、ASPEC-I の目標である統語や単語綴りの範囲では、76% が検出できた。

1. ま え が き

われわれが文章を書くとき、その文がつねに正しいとは限らない。誤字のような誤りから、文法や意味の上の誤りまで、種々の誤りが生じうる。したがって、このような文の中の誤りを、自動的に検出してくれるシステムがあると便利である。こうした、誤りを含む文章の計算機による処理システムは、単語綴りの誤りの検出や文体上の特徴を指摘してくれる UNIX のシステム^{1),2)} や、文法ならびに文体的な誤りを検査してくれる EPistle³⁾ などがある。

これらのシステムは、いずれも母国語で書いた文(英文)の誤り検出システムである。ところが、外国語で作文するときには、母国語で書く場合とは異なった種類の誤りも犯しやすい。このことは、英語についていえば、英語学の分野でいろいろと指摘されているが(文献 4), 5) など、計算機による自然言語処理の立場からは、あまり研究がなされていない。

また、現在の技術では、いろいろな分野の文章にたいして、意味や文脈処理を、人間が行うのと同じレベルで行うことはむずかしい。しかし、外国語で作文する場合には、意味や文脈レベルの誤りに加えて、比較的、初歩の誤りも犯しやすい。したがって、高度な処理をシステムが行わなくても、ある程度、役にたつと考えられる。

そこで、われわれは、日本人の書く英作文の添削を支援するシステム ASPEC-I (an Advisory System

for English Composition I) を作成した。本システム (ASPEC-I) は、入力された英文の誤り、主として統語的な誤り、の有無をみつけ、誤りがあればエラーメッセージを表示するシステムである。ASPEC-I は、文献 6) で提案した手法を一部に組み入れた、実際に動くシステムである。

本論文は、この ASPEC-I について述べている。まず、2 章において、ASPEC-I がどんな方法でエラーメッセージを表示するかについて説明する。次に、3 章で、ASPEC-I が検出できる誤りの例を示す。4 章では、ASPEC-I について行った二つの実験について報告する。

2. 誤り検出の方法

2.1 対象とする誤りの範囲

英文を書き慣れていない日本人の英作文を、米国人に添削してもらった場合を考えてみよう。そのなかには、単語綴りの誤りもあるかもしれないし、冠詞の使い方、動詞の使い方、さらには文の内容に関する誤りもあるかもしれない。こういった、いろいろな種類の誤りを、人間は同時に検出できる。しかし、誤り検出のための計算機プログラムを設計するためには、まずこれらの多種多様な誤りを分類しなければならない。そこで誤りを、次のように四つに分類した。

(1) 単語綴りの誤り

【例】 the extention of a railway line. (extension が正しい。extention なる単語はない)

(2) 統語的な誤り

【例】 I expects him to go. (expect が正しい)

(3) 意味的な誤り

【例】 The clock believes that this is so. (時計は

† ASPEC-I: An Error Detection System for English Composition by ATSUO KAWAI, KŌKICHI SUGIHARA and NOBORU SUGIE (Department of Information Science, Faculty of Engineering, Nagoya University).

†† 名古屋大学工学部情報工学科

物質なので信じていけない)

(4) 事実と相違する誤り

[例] The IUPAC names of ketones are derived from the name of corresponding alkane by replacing the final -one by -e. (-one と -e が逆. この誤り検出には, 化学の専門的知識が必要)

こうした誤りをすべて検出する計算機システムを作るためには, システムが, 辞書, 文法をもっているだけでなく, 十分な意味, 文脈処理を行い, いろいろな知識を組み合わせる推論する機能がなければならない。しかし, こうしたシステムを作ることは, 現在の自然言語処理技術ではむずかしい。

そこで, 本論文では, 単語綴りの誤りと, 統語的な誤りを考察の対象とする。その理由は, 第一に, すでに研究が進んでいる(1)の語彙情報のみを注目した誤り検出^{1),2)}の次になすべき研究段階であるからである。第二に, 誤り検出システムは, 書き手の気がつかないいくつかの誤りを検出できれば, すべての誤りを完璧に検出できなくても, 実用的な役にたつと考えられるからである。

2.2 ASPEC-I の英文処理の流れ

前節で述べたように, ASPEC-I は綴り字の誤りと, 統語的な誤りを検出する。そのための英文処理の流れを図1に示す。

綴り誤りは, ①で検出される。つぎの②の構文標準化⁶⁾は, 一般の文法規則では取り扱いにくい構文(たとえば, He said nothing as to hours.)にたいしてのみ, 変換(標準化)を行って, 基本的な構文(He said nothing as~to hours.*)に書き換える。この基本的な構文は, 補強文脈自由文法で処理できる。もし, 補強文脈自由文法のみを使って数多くの例外的な構文を処理しようとする, 文法のサイズが爆発し, 見通しも悪くなってしまう。構文標準化法は, この難点を克服するためのものである。つぎに, 補強文脈自由文法を利用して, 統語的な誤りをみつける(③~⑦)。これについては, 2.3節, 2.4節において詳しく述べる。

2.3 補強文脈自由文法を使った英文法

補強文脈自由文法は, 一つの記号が他のどんな記号に置きかえられるかという規則(生成規則)と, その

* この文において, 書き換える前では, 前置詞が二つ続く(as to)構文であったが, これを一つの群前置詞とみなすことにより, 標準化する。前置詞が二つ続く構文は, 一般的ではない。

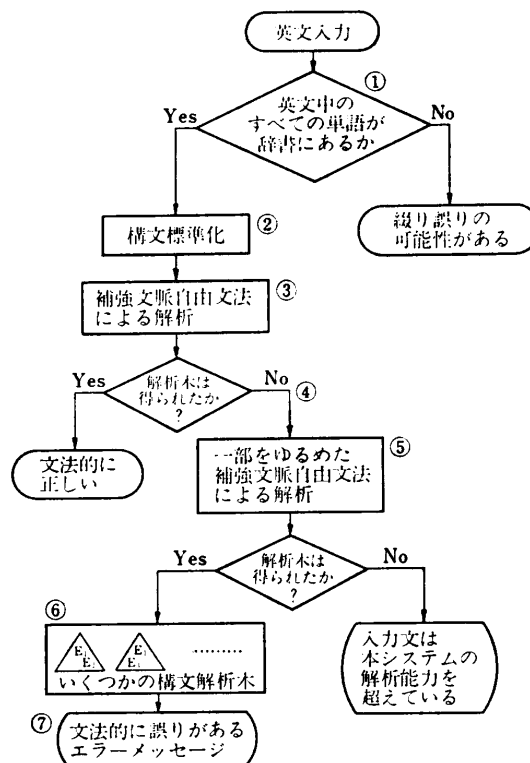


図1 英文解析の流れ図

Fig. 1 Flow chart of sentence analysis.

⑥で, それぞれの三角形は, 1個の解析木を表し, E_nは, 誤りの検出された場所およびその内容を表している。

生成規則をどんなときに適用するかを決める部分(生成規則の制御部分)から成り立っている。ASPEC-Iの補強文脈自由文法の制御部分は, LISPで書かれたいくつかの関数からできている。

I have an apple. (例文1)

の解析に使う補強文脈自由文法の一部を, 図2に示す。最初の生成規則は, 名詞句が, 冠詞と, それ以外の部分(#NSTG)から成ることを表している。また, (KA-A-AN)などは, この規則の適用を制限するLISP関数である。

関数“(KA-A-AN)”は, 冠詞が誤った使われ方をしていると, 生成規則の適用を禁止する。冠詞の使われ方の当否は, たとえば, 名詞句の一部である“#NSTG”の最初の単語が母音で始まる時, 冠詞を

```
(NSTG→(LN #NSTG); (AND(KA-A-AN) (WAGREE9)...))
(#NSTG→NOUN ; NIL
(LN →ARTICLE ; NIL
)
```

生成規則 生成規則の適用を制御する部分

図2 補強文脈自由文法の一部

Fig. 2 An excerpt of the augmented context-free grammar.

```
(I HAVE A APPLE /.)      → 入力文
+ANALYSIS#1: NIL        ③
"WE CAN NOT PARSE THIS SENTENCE(FIRST ANALYSIS)."  
+E-ANALYSIS#1: NIL      ⑤
XXXXXXXXXXXXXXXXXXXXX  
1 (得られた解析木の数) ⑥  
"THE FOLLOWING ERRORS ARE FOUND."  
(ERROR1)  
(DISAGREEMENT OF (A) AND (APPLE)) ⑦
```

図 3 エラーメッセージの出力例
Fig. 3 An example of error message.

③～⑦は、図1と対応している。

表す“LN”が単語“a”であれば、冠詞の使われ方が誤っていると判定する。

2.4 ゆるめられた補強文脈自由文法

正しい英文のみが入力されるのであれば、前節の方法でよい。しかし、例文2

I have a apple. (例文2)

のように、統語的な誤りを含む文が入力されれば、関数“(KA-A-AN)”が、生成規則の適用を禁止して、構文解析木を得ることができない。たんに解析木が得られないという情報のみでは、システムの文法が未熟なのか、それとも英文のほうに誤りがあるのかわからない。また、英文が誤っているにしても、どこに誤りがあるかわからない。

そこで、ASPEC-Iでは、補強文脈自由文法をゆるめることにより、誤りをみつける。すなわち、生成規則の制御部分の条件をゆるやかにする方法である。たとえば、“(KA-A-AN)”は、冠詞の部分があると“an”であっても、生成規則の適用を許すことにする。

ここで、すべての制御規則をゆるめると、非常に多くの解析木ができてしまい、どこに誤りがあるかわからない。そこで、文献4)、5)などを参考にして、制御規則を無駄な解析木を生成させない規則と、誤りをみつけるための規則の二つに分けた。そして、誤りをみつけるための制御規則だけを、ゆるめないと生成規則が適用されない状況になったときだけゆるめて、2度目の解析を行う(図1の⑤)(ゆるめないままで適用できる生成規則は、ゆるめない)。

そして、ゆるめた制御規則を使った場合にのみ、使ったことを解析木の中に記録しておく(図1の⑥のEn)。この記録をもとに、エラーメッセージを表示する(図3)。ゆるめることによる解析木の増加はそんなに多くない(4章表1参照)ので、誤り箇所はすぐに特定できる。もし、いろいろな形の解析木ができて誤り箇所の特定ができないときには、各解析木ごとにゆるめた場所(図1の⑥)を表示させ、誤り箇所を決定する。

この手法の基本的な考え方は、EPISTLE³⁾と同じである。

3. ASPEC-I の概要

ASPEC-Iは、名古屋大学大型計算機センターのFACOM M-382上に作られている。ASPEC-Iの誤り検出部分は、UTILISP版の拡張LINGOL⁷⁾を拡張して作成した。

文法は、文献8)、9)を参考に作成し、日本人が犯しやすい統語的誤り(可算名詞と不可算名詞の使い分け、関係代名詞の使い方、形容詞の限定的用法と叙述的用法の使い分け、冠詞の使い方)の検出のために、生成規則の制御用の関数を多くつけ加えている。生成規則の数は402個(各規則は、1個の非終端記号が、たかだか2個の非終端記号または終端記号に分かれるという様式で表されている)、生成規則の制御用の関数の数は81個(このうち、誤りをみつけるための制御規則は35個)である。

また、辞書はおもに4章で述べる実験のために作成したので、単語数は、まだ約2,300語である。

ASPEC-Iが検出できる統語的誤りの例を示す。

(1) 主語と動詞の間の数の不一致

They *does* not generally walk. (do が正しい)
Going to school *are* important. (is が正しい)
Who *know* the book? (knows が正しい)
I have two uncles who *is* very kind. (are が正しい)

(2) 名詞句内における誤り

I have *a* apple. (an が正しい)
Do you have *a bread*? (a loaf of bread などが正しい)
How *many* money do you have? (much が正しい)
I bought *this* two fish. (these が正しい)
Such *the* people can do it. (a が正しい)
You looked for the *tall* five people. (five tall または tallest が正しい)
House is on the hill. (the, another, my などを house の前につけた文が正しい)
Only *a* she can do it. (a を除いた文が正しい)

(3) 形容詞の用法の誤り

An *old fat* lady looks ugly. (fat と old の語順が逆)

It is *an alive* snake. (a live などが正しい。叙

表 1 校閲を終えた英文の構文解析
Table 1 Sentence analyses of revised sentences.

		解析率	時間切れ	平均解析木数/文	平均解析時間/文
情報処理学会 (1文あたり平均 19.3 語)	ゆるめない文法	65%	4%	9.9 個 (65 文平均)	2,790 msec (65 文平均)
	ゆるめた文法	83%	5%	13.1 個 (83 文平均)	3,398 msec (83 文平均)
Scientific American (1文あたり平均 21.4 語)	ゆるめない文法	49%	0%	7.1 個 (49 文平均)	2,365 msec (49 文平均)
	ゆるめた文法	65%	1%	14.3 個 (65 文平均)	3,525 msec (65 文平均)

述的に用いる形容詞を限定的に用いた誤り)

(4) 動詞の用法の誤り

This medicine must *cures* your cough. (助動詞の後にくる動詞は原形)

Please *explain* me the meaning.* (動詞 explain は二重目的語をとらない)

(5) その他

I met *she* on the street. (her が正しい)

I do not like people *which* lose their temper easily. (先行詞が人のときは who が正しい)

I am *boys*. (主語と補語の間の数の不一致. a boy が正しい)

4. 実験

ASPEC-I が、科学技術について書かれた英文にたいして、どれだけ有効であるかを調べるための実験を行った。まず、4.1 節では、すでに十分な校閲を終えた英文にたいして、ASPEC-I の構文解析能力がどれくらいあるかを調べた。つぎに、4.2 節では、まだ添削をされていない英作文を対象に、ASPEC-I の誤り検出機能がどれくらいあるかを調べた。

4.1 ASPEC-I の文法の構文解析能力

ASPEC-I の文法を使うと、どれくらいの割合で、適切な構文解析木を得ることができるかを調べた。

[実験に使った英文]

英米人の書く英文として、Scientific American に記載された計算機科学に関する五つの論文 (Vol. 245, No. 4, p. 136; Vol. 245, No. 6, p. 102; Vol. 246, No. 1, p. 112; Vol. 247, No. 4, p. 101; Vol. 248, No. 3, p. 36) から、それぞれ先頭の 20 文ずつ、合計 100 文、および日本人の書く英文として、情報処理学会誌の英文アブストラクト (*J. Information Processing*, Vol. 5, No. 2, pp. 113-118 (1982)) の 20 編から、それぞれランダムに 5 文ずつ、合計 100 文。ただ

し、数式を含む英文は除いて選んだ。また、英文中に存在する“()”は除去した。

[結果]

得られた結果を表 1 に示す。表 1 において、解析率は、英文に対して解析木が得られ、かつそのなかに適切な解析木が含まれている場合に、解析が行われたとして計算した。また、時間切れとは、計算機の作業領域として約 2MB を使ったとき、CPU-time が 40 秒を超えた場合である。

[考察]

制御規則をゆるめないと解析できない英文も存在する。この原因のほとんどは、4.2 節に述べる可算/不可算名詞の取扱いに関連している。

文法をゆるめても、解析木の数はそんなに増加していない。ただ、この一つの原因は、ホーンビーの動詞パターン⁹⁾をゆるめていないためである。これは、ゆるめてしまうと、比較的長い英文(単語数 40 語程度)の解析時間が非常に長くかかるからである。実験で使った英文 200 文中、文の長さが短い 40 文を選んで動詞パターンをゆるめたら、制御規則をまったくゆるめないときにくらべて、解析木は平均して 6 倍になった。

つぎに、得られた結果を EPISTLE³⁾ のシステムと比較してみる。EPISTLE では、商業用の手紙文から英文を採っており、1 文あたりの単語の数は平均すると、19 である。したがって、対象とする英文の長さは、同程度である。解析率は 64% であるので、ASPEC-I が英米人の書く英文に対して得た結果 (65%) と似ている。また、EPISTLE では、一つの英文にたいして、一つの構文解析木しか得られない割合が、43% もある。これにたいして、ASPEC-I が作り出す解析木の数が多く原因は、副詞や前置詞句がどこにかかるかについて多くの可能性について調べているためや、“,” の処理についても、意味処理を導入していないため、曖昧性が残っているためなどである。

* ホーンビーの動詞パターンで検出するため、長い英文では、この種の誤りをみつけるのはむずかしい。4.1 節参照。

これにたいして、日本人の書く英文では、英米人の書く英文よりも、高い解析率 (83%) を得ることができた。したがって、この点から、母国語としない人が書く英文のための添削システムのほうが、作成が容易であるといえそうである。時間切れについては、生成規則の構成を工夫したり、ASPEC-I のデータ構造を改良すれば、減らすことが可能である。

英文が解析できなかったおもな理由は、生成規則の数が十分でなかったからである。したがって、倒置、省略、その他の理由で、システムの生成規則を使って表せない文は、構文解析ができない。しかし、単純に生成規則を増加してしまうと、誤りをみつけられなくなる可能性がある。これは、誤りを含む英文にたいしても、英文を書いた人が意図しない形の構文解析木が作られてしまうからである。また、作られる解析木の数が増加して、計算時間や記憶量の点で、カバーできない可能性がでてくる。

ASPEC-I では、補強文脈自由文法を用いて、英文法を記述している。しかし、関係代名詞節の記述には、補強文脈自由文法では、不十分である。これにたいして、文献 10) には、拡張文脈自由文法が提案されている。こうした文法を使うと、システムの文法能力が、向上することが考えられる。

4.2 ASPEC-I の誤り検出の性能評価

ネイティブスピーカーが、英文の誤りを指摘してくれるのと同じような性能を誤り検出システムがもつことができれば、理想的である。そこで、本節では、日本人の書く英作文を使って、誤り検出の能力を評価する。すなわち、ネイティブスピーカー (名古屋大学文学部外国人教師) に、英文を添削して誤りをみつけてもらい、このうち ASPEC-I によって検出できる誤りがどれだけあるかを調べる。そして、システムの性能を評価し、今後の発展方向についても考える。

[英作文の方法]

題材: 「サイエンス (Vol. 11, No. 12, pp. 102-104 (1981, 12))」から採用 (図 4 に示す)

被験者: 教官 2 名, 大学院生 6 名

所用時間: 平均約 90 分

辞書、英語の参考書などを参照してもよいこととし、専門用語 (図 5 に下線で示す) には、あらかじめ英訳を与えた。ただし、他人とは相談しないこととした。

[ネイティブスピーカーによる添削の方法]

一口に英作文の誤りといってもいろいろある。そこ

アナログ画像をデジタル画像に変換する方法の 1 つに放送用テレビを用いるものがある。テレビカメラでは、レンズ系で形成された画像が多数の水平方向の走査線に分解される。各走査線に沿った輝度の変化は、連続的な電気信号として送られる。それゆえ、テレビ画像は部分的にデジタル化されている。すなわち、垂直方向には離散的な情報になっているが、水平方向では離散的になっていない。

画像におけるこの情報の表現形態から進んで、デジタルコンピュータへ入力できる形態にするには、もう 1 つの段階、すなわち各走査線を一組の離散的な値に分解する段階が必須である。こうして、画像の全面が画素の配列で表現される。画素 (picture element) はピクセル (pixel) とも呼ばれる。各画素の明るさは、原画像の対応する領域の平均的な明るさに等しい。

図 4 英作文の題材

Fig. 4 Material of English composition.

で、誤りの検出とともに、その分類を、2.1 節および以下に述べる基準に従って行ってもらった。2.1 節の分類と重複する部分もあるが、添削の都合上、(6)~(9)を追加した。1~4(2.1 節)の誤りのどれかに分類され、かつ(6)~(9)にも分類される誤りは、(6)~(9)に含めた。

(6) 冠詞の使用の誤り

[例] I think there is the letter for you. (一般的には、“a”が正しい)

(7) 前置詞の選択の誤り

[例] love to a girl. (“女の子への愛情”の意味で使うときには“for”が正しい)

(8) 文体的な誤り

冗長な表現、否定の使いすぎ、文章が長すぎる、など

(9) 語彙の選択の誤り

文の意味は理解できるが、よりの確に表現するためには、ある単語(列)を、別の単語(列)に置きかえたほうがよい場合

[例] the corresponding region→the corresponding area

被験者による英作文と、その添削例を、図 5 に示す。

[ASPEC-I による検出]

英作文を ASPEC-I に入力し、エラーメッセージが出力されたとき、誤りが検出されたとした。ただし、ASPEC-I の文法は、省略を含む等位接続詞・比較の

- 1 One of ^{the(6)} methods that we use to convert ^{an(6)} analogue image into ^{a(6)} digital image is ^{through(7)}
 (6)
 [a] broadcast television.
- 2 The television ^{camera(1)} camera divides the image constituted by ^{a(6)} system of lenses
 into ⁽²⁾ many horizontal scan lines.
- 3 Variation of brightness along each scan line is sent ^{as(7)} ^{a(6)} by the continuous
~~electric~~ ^{electrical(9)} signal.
- 4 So, some parts of ^{the(6)} television image ^{are(2)} is digital.
- 5 Although it is discrete in the vertical direction, it is indiscrete in the
horizontal direction.
- 6 In order to convert this ^{representation(9)} ^{the(6)} expression of image into the ^{representation(9)} expression used to input
^{a(6)} digital computer, one more step is necessary.
- 7 It is ^{to divide(8)} dividing each scan line ^{into(7)} in a set of digital data.
- 8 Then all ^{part(2)} ^{the(6)} parts of image are expressed ^{as(7)} as a matrix of picture elements.
- 9 Sometimes we call the picture element ^{a(6)} a pixel.
- 10 ^{The(6)} Brightness of each picture element is equal to average brightness of ^{the(6)}
^{to(7)} the area that corresponds with the original image.

図 5 英作文とその添削例

Fig. 5 An example of English composition and its correction.

【凡例】

↑: a, the などを挿入する必要がある

[a]: a を取り除く

(6): 誤りの分類番号

parts: 誤りの原因となる単語。↑すぐ上に示した単語で置き換えると正しい英文となる。scan line: あらかじめ与えた専門用語

取扱いが不十分であるため、その文型が現れたとき (たとえば、Vertical information is discrete but horizontal is not.) には、ASPEC-I で扱える形 (Vertical information is discrete but horizontal information is not discrete.) まで書き換えて入力した。

また、一つの英文中に複数個の誤りがあるときには、一般に複数のエラーメッセージが表示される。しかし一カ所を修正しないと、別の誤りがみつけれないこともある。(たとえば、一つの文に綴り誤りと統語的な誤りがあったとする。このときは、綴り誤りを修正して初めて構文解析が可能になり、統語的誤りを

みつけることができる。) このときは、修正した英文を入力して、順次、エラーメッセージを得た。

【結果】

71個の英文(1人7~11文; 合計8人)にたいする、ネイティブスピーカーによる添削結果と、ASPEC-I による誤り検出の結果を表2に示す。分類番号4の誤り(事実と相違する誤り)の指摘には、画像処理に関する十分な知識が必要であるため、添削の範囲からは除外した。また、6の誤り(冠詞の使用の誤り)の割合が多いので、これをさらに下位分類した(表3)。

ASPEC-I は、ネイティブスピーカーのみつけた誤

表 2 人間による添削と ASPEC-I による誤り検出の比較
Table 2 Comparison of error detection (native speaker vs. ASPEC-I).

誤りの分類	1 単語綴り	2 統語	3 意味	6 冠詞	7 前置詞	8 文体	9 語彙の選択
人間が添削してみつけた誤り	7	7	0	46	10	8	24
システムが検出した誤り	7	4	0	27*	0	0	0

* 27 個以外に、誤りではないのにエラーメッセージを出した箇所が、10 個あった。

表 3 冠詞の使用の誤りの下位分類 (△; 統語的誤りに分類される誤り)
Table 3 Subclassification of article usage errors.

	不定冠詞と 定冠詞の混同	"a" と "an" の混同	冠詞が必要 なのにない	冠詞が不要 なのにある	その他
人間が添削してみつけた誤り	9	3 [△]	24 [△]	9 [△]	1
システムが検出した誤り	0	3	24	0	0

り全体の 37% を検出した。また、ASPEC-I の目標である、綴りと統語の誤り (表 3 では、△で示す) だけに注目すると、76% を検出した。

【考察】

2 (統語的誤り) において、エラーメッセージが表示されたのは、図 5 の英文 4・8 にみられる、主語と動詞の間の数の不一致や、名詞句内における数の不一致の誤りである。ASPEC-I が検出できなかった誤りとしては、図 5 の英文 2 の誤り (is→into) がある。この文は、構文解析木が得られなかったため、どこに誤りがあるかは、みつけられなかった。

6 (冠詞の使用の誤り) では、次の三つの点に注目して誤りを検出している。① "a" と "an" の混同。② 可算名詞が単数のときに、冠詞があることの確認。ただし、冠詞にかわる単語 (my, another など) があるときは、除外する。③ 名詞句内の数の不一致 (例: a boys)。

したがって、これに関する誤りは、検出できた。

ただ、可算名詞としても不可算名詞としても使われる名詞 (例: system) では、冠詞の使用の検査を正確に行うのはむずかしい。なぜなら、不可算名詞として使われていたら、冠詞はなくてもよいからである。可算か不可算かの判断を、システムが行うのはむずかしい。そこで、誤りの可能性があるときには、すべてエラーメッセージを表示している。したがって、今回の実験でも、誤りではないのにエラーメッセージを出してしまう箇所は、10 個あり、いずれも可算/不可算名詞の取扱いに関連がある。このうち、熟語として (例: in advance of) 使われる箇所が 4 個、そうでない名詞 (variation, representation など) 6 個である。

定冠詞 (the) と不定冠詞 (a, an) の使い方の検査には、一般的に、文脈・意味理解の面からのアプローチが必要である。したがって、ASPEC-I では、検出できていない。また、冠詞が不要であるのにつけた誤りとして、慣用的に冠詞なしで使われる "broadcast television" が冠詞とともに用いられた誤り (図 5 の英文 1) が五つあった。

7 (前置詞の選択の誤り) は、英作文において、冠詞の使用の誤りと同様、誤りやすいことが指摘されている⁴⁾。しかし、前置詞の選択には意味もからんでくるので、一般規則として取り扱うのはむずかしく、ASPEC-I では検出できていない。英作文のときに、人間が辞書で動詞と前置詞の組合せを調べるといった働きを、システムにくみこむことができれば、前置詞の誤り検出も可能になると思う。

また、誤りといっても、重大な誤りから、ネイティブスピーカーにとって何となく奇妙にみえる程度の軽い誤りまで、いろいろなレベルがある。これらのレベルを区別して表示できるようにすることも必要である。

5. むすび

英作文の添削を支援するシステム、ASPEC-I について報告した。ASPEC-I は、英作文における綴り誤りと統語的誤りを検出する。統語的な誤りは、補強文脈自由文法の一部をゆるめることにより、検出している。

ASPEC-I の文法を使うと、十分な校閲をした科学技術論文の英文にたいして、どれくらいの割合で構文解析が可能であるかを調べた。英米人の書く英文では、65% で、ネイティブスピーカーを対象とした、

EPISTLE³⁾ とほぼ同じであるが、日本人の書く英文では、83% と割合高い値が得られた。

また、日本人の書いた、添削を受けていない英文を使って、システムの性能評価実験を行った。ASPEC-I は、誤り全体の 37% を検出した。ASPEC-I の目標であった綴り、統語的誤りの範囲では、76% が検出できた。また、英作文の誤りの中で、冠詞の使用の誤りの割合が高いことが確認できた。

今後のシステムの性能向上のためには、4章で述べたように、文法の改良および、意味処理を用いた検査の機能の追加が必要である。

謝辞 英作文の添削をしてくださった、名古屋大学文学部の M. Horne 先生、UTILISP 版の拡張 LINGOL の利用に際してお世話いただいた、電総研田中穂積氏 (現在、東工大)、東大計数工学科白井英俊氏 (現在、玉川大)、名大計算機センター長谷川明生氏に感謝します。また、本研究の一部は、文部省科研費 57580022 により行われ、プログラムの実行には、名古屋大学大型計算機センターを利用した。

参 考 文 献

- 1) Macdonald, N.H.: The UNIXTM Writer's Workbench Software: Rationale and Design,

Bell Syst. Tech. J., Vol. 62, No. 6, pp. 1891-1908 (1983).

- 2) Gringrich, P. S.: The UNIXTM Writer's Workbench Software: Results of a Field Study, *Bell Syst. Tech. J.*, Vol. 62, No. 6, pp. 1909-1921 (1983).
- 3) Heidorn, G. E., Jensen, K., Miller, L. A., Byrd, R. J. and Chodrow, M. S.: The EPISTLE text-critiquing system, *IBM Syst. J.*, Vol. 21, No. 3, pp. 305-327 (1982).
- 4) Peter Milward: English—Right and Wrong (英語の正誤診断), 研究社, 東京 (1972).
- 5) Swan, M.: *Practical English Usage*, Oxford University Press, Oxford (1980).
- 6) 河合, 杉原, 杉江: 英文の誤り検出のための構文標準化法, 信学会論文誌, Vol. J66-D, No. 5, pp. 511-518 (1983).
- 7) 電総研推論機構研究室: 拡張 LINGOL (1978).
- 8) Sager, N.: *Natural Language Information Processing*, Addison-Wesley, Reading (1981).
- 9) AS ホーンビー (伊藤健三訳): 英語の型と語法, オックスフォード大学出版局, 東京 (1977).
- 10) 西田, 堂下: 自然言語解析のためのプログラミングシステム COMPLAN について, 情報処理学会論文誌, Vol. 23, No. 4, pp. 396-405 (1982).

(昭和 59 年 2 月 8 日受付)

(昭和 59 年 6 月 19 日採録)