DaHae:일한 기계번역을 위한 일본어 형태소 분석기

여상화, 정한민, 장원, 김태완, 황도삼, 박동인

국어공학센터/시스템공학연구소

DaHae:Japanese Morphological Analyzer for Japanese to Korean Machine Translation

Sanghwa Yuh, Hanmin Jung, Won Chang, Taewan Kim, Dosam Hwang, Dongin Park

Center for Korean Language Engineering/Systems Engineering Research Institute

요약

일본어는 한자, 히라가나, 가다가나 등 다양한 종류의 문자를 사용하며 이들의 혼용 비율이 매우 높아 띄어쓰기를 하지 않아도 문서의 가독성을 유지한다. ICOT 사전, EDR 사전, ATLAS I/JK사전 등 기존의 전자 사전에서 복합 자종의 표제어가 차지하는 비율(한자+히라가나의 표제어 제외)은 평균 8.8%로 그 수가 매우 작다. 따라서, 문장 내에서 자종의 변화는 단어를 구분하는 하나의 delimiter로 이용될 수 있다.

본 시스템에서는 형태소 분석의 전단계로 전처리기를 두어 자종정보(character type information)에 의한 fragment 분리 및 예외 단어, 정형표현 처리를 수행하며 각 fragment의 형태소 분석 방법을 제시한다. 형태소 분석기는 전처리기의 처리 결과를 입력 받아 각각의 fragment를 전처리기가 제시한 분석 방법에 따라 분석하여 입력 문장의 가능한 모든 분석을 추출한다. 이 방법은 불필요한 사전 탐색과 접속 체크 회수를 줄여 분석 성능을 향상시킨다.

1. 서론

일본어는 띄어쓰기를 하지 않지만 한자의 혼용 비율이 매우 높고 외래어나 의성어 등은 가다가나로 표현하는 등 여러 종류의 문자가 이용되기 때문에 띄어쓰기를 하지 않아도 문

장을 이해하는데 어려움이 없다.

표 1-1은 당 연구소와 일본 후지쯔가 공동 개발한 ATLAS I/JK의 번역 사전과 일본 Kyoto 대학에서 개발한 일본어 형태소 분석기 JUMAN V2.0[松本裕治94]에서 형태소 분석 사전으로 이용하는 EDR사전, 그리고 ICOT에서 개발한 형태소 분석기가 사용하는 사전에 수록된 표제어의 유형을 조사한 결과이다. 복합 자종 표제어의 비율은 평균 16.7%이며 이중에서 한자 다음에 히라가나가 나오는 표제어를 제외하면 평균 8.8%로 매우 적다. 이는 문장 내에서 자종의 변화를 단어를 구분하는 delimiter로 볼 수 있음을 의미한다.

표 1-1 기존 사전의 표제어 유형 분석

구분	ICOT	EDR(JUMAN V2.0)	ATLAS I/JK
1. Uniq한 전체 표제어 수	119,824	166,651	55,391
2. 같은 자종의 표제어 총수	101,117(84.4%)	142,569(85.5%)	44,350(80.1%)
-한자	76,688	76,339	34,980
-히라가나	16,344	57,760	4,750
-가다가나	8,044	8,364	4,571
-숫자	2	10	10
-영문자	38	45	20
-기호	1	51	19
3.복합 자종의 표제어 수	18,707(15.6%)	24,082(14.5)	11,041(19.9%)
-한자 다음에 히라가나	8,437(7.0%)	13,445(8.1%)	4,667(8.4%)
-기타	10,270(8.6%)	10,637(6.4%)	6,374(11.5%)

주) 1. ATLAS I/JK 사전: KEF Code에서 KS C-5601로 코드 변환한 후 미정의 문자를 포함한 표제어를 제외함

2. JUMAN 사전: 고유명사를 제외한 표제어를 대상으로 함

자종의 변화를 단어를 구분하는 delimiter로 볼 수 있다 하여도 일본어 문장에서 이 현상이 충분히 나타나지 않는다면 이를 이용한 기대 효과는 미미할 것이다. 그림 1-1은 일본 신문에서 추출한 1,489,175문자 중 자종별 분포를 보여 준다[HIROSHI80].

한자	히라가나	가다가나	숫자	기호
43.4	28.0	8.1	9.8	9.2

↑ 영문자0.6

그림 1-1 일본 신문에서의 자종별 분포

또한, 표 1-2는 [長尾 眞]의 전자 문서(화일1)와 network을 통해 수집한 일본어 text 뭉치(화일2)에서 문자의 혼용 비율을 조사한 결과이다. 이들로부터 일본어에서는 한자의 혼용 비율이 매우 두드러짐을 알 수 있다.

표 1-2 일본어 문서에서의 자종별 분포

구분	화일 1	화일 2	평균
문장의 수	2,379	58,002	60,381
File 크기(byte)	260,420	5,904,948	6,165,368
문자 수	**129,827**	**2,972,189**	**3,102,016**
-한자	37,679	949,994	987,673(31.8%)
-히라가나	77,976	1,528,051	1,606,027(51.8%)
-가다가나	5,345	174,624	179,969(5.8%)
-숫자	406	8,537	8,943(0.3%)
-영문자	1,462	24,824	26,286(0.8%)
-기호	6,959	286,159	293,118(9.4%)
-기타(빈칸,CR, Tab 등)	3,258	37,622	40,880

본 논문에서 제안하는 일본어 형태소 분석기는 일본어의 자종정보를 이용하여 입력 문장을 여러 개의 fragment들로 분리한 후, 각각의 fragment를 분석하여 입력 문장의 가능한 모든 분석을 추출한다.

2장에서는 기존의 연구를 살펴보고 3장에서는 본 논문에서 제안한 형태소 분석기 DaHae의 구성을 설명한다. 4장과 5장에서는 분석 알고리즘을 설명하고 수행 예를 보인다.

2. 기존의 연구

기존의 일본어의 형태소 분석에서 자종 정보를 이용하려는 시도는 [HIROSHI80]과 [MARUYAMA88][강석훈95] 등에서 찾아 볼 수 있다. [HIROSHI80]는 표2-1과 같이 자종에 따른 분리 행렬을 구성하여 0이면 결합시키고 1이면 분리하는 방식을 취한다.

표 2-1. 자종의 결합에 따른 분리 표

앞 \ 뒤	한자	히라가나	가다가나	영문자	숫자	기호
한자	0	0	0	1	1	1
히라가나	1	0	1	1	1	1
가다가나	0	1	0	0	0	1
영문자	0	1	1	0	0	1
숫자	0	1	0	1	0	1
기호	1	1	1	0	0	0

위의 표로 분리되지 않는 조사, 조동사, 부사 등은 형태소 분리와 품사가 지정된 기분석 테이블을 이용하여 분리한다. 그림 2-1에서 box로 되어 있는 부분이 기분석 테이블을 이용하여 분리된 부분이다.

입력 문장 : COLING80が東京の都市センターホールで開催された.

분석 결과 : COLING80/が/東京/の/都市センターホール/で/開催さ/れ/た/.

그림 2-1. 분석 예

[MARUYAMA88]는 자종의 패턴과 Action의 쌍으로 이루어진 테이블에 의해 입력 문장을 1차적으로 분리하고, 분리된 각 fragment 끝에서 "を", "あたって" 등 절대적인 segmentation boundary를 제공하는 문자열을 검색하여 이들을 2차로 분리해 낸다. 최종적으로 분리된 각각의 fragment에 대해서 최장일치법을 적용하여 분석한다.

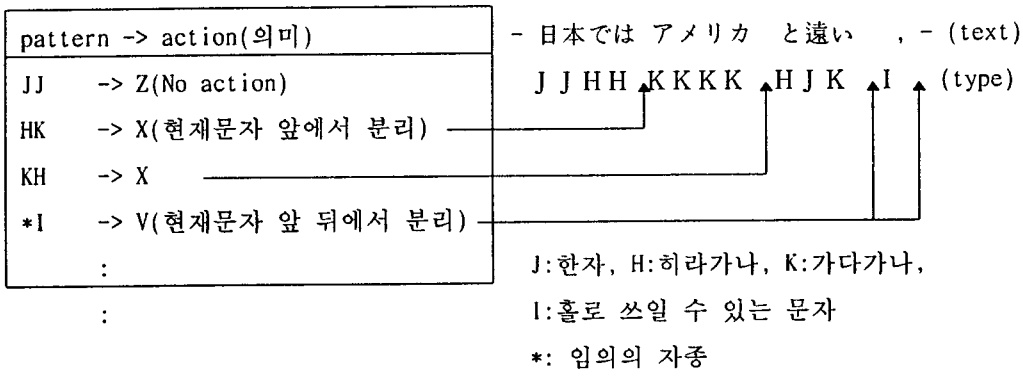


그림 2-2. 자종 pattern에 의한 분리의 예

자종을 형태소 분석에 이용하는 경우 표 1-1에서 제시한 바와 같이 복합 자종의 표제어의 처리 문제가 대두된다. 이를 해결하지 못하면 한 단어가 둘로 분리되는 문제가 발생하나 [HIROSHI80][MARUYAMA88][강석훈95]에서는 이 문제에 대한 구체적인 언급이 없다.

3. DaHae 시스템의 구성

본 논문에서 제안한 형태소 분석기, DaHae의 구성 및 처리 흐름은 그림 3-1과 같다. 일본어 코드 변환기를 사용하여 입력 문서를 일본어 EUC(Extended Unix Code)코드로 변환하여 처리하므로 입력 문서의 일본어 코드는 구JIS(JIS X-0208:1978), 신JIS(JIS X-0208:1990), Shift JIS(Microsoft JIS), EUC, NEC JIS, KS C-5601 등 제한이 없다.

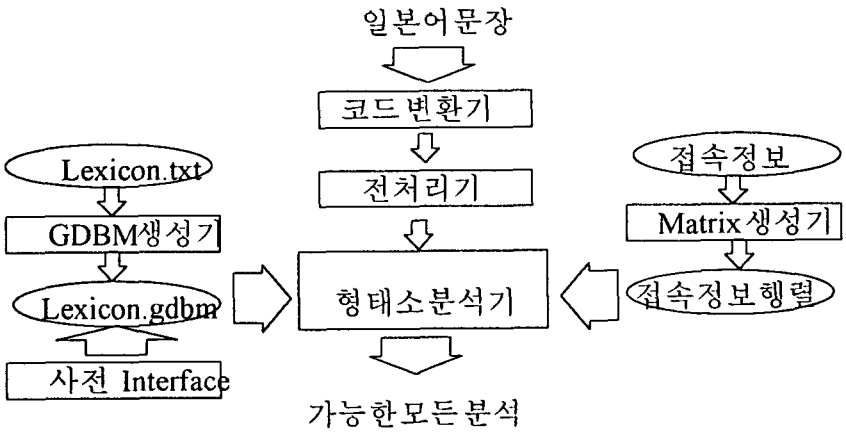


그림 3-1. DaHae의 구성

3.1 일본어 코드 변환기

일본어의 코드는 구JIS, 신JIS, Shift JIS, EUC, NEC JIS 등으로 매우 다양하다 [Lunde92]. DaHae는 일본어 코드 변환기 jconv V2.3[Lunde92]과 본 센터에서 개발한 KS 완성형 코드와 일본어 EUC간의 코드 변환기(ks2euc, euc2ks)를 이용하여 입력 문서를 일본어 EUC 코드로 변환하여 분석한다.

3.2 전처리기

사전 표제어 중의 약 9%(표 1-1)는 복합 자종으로 구성된 표제어로 자종에 의한 기계적인 분리는 한 단어를 잘못 분리하는 오류를 범하게 된다. 예를 들어 "けしコム(지우개)"라는 표제어를 [HIROSHI80]에 제시한 방법대로 분리하면 'けし"와 "コム"로 잘못 분리된다.

전처리기는 자종에 의한 일반적인 fragment 분리와 이에 따른 복합 자종의 표제어 처리 이외에도 다음과 같은 역할을 수행한다.

1) 구점(。:0xA1A3)을 이용하여 문장 단위로 분리한다.

2) 입력 문장에서 분석 대상이 아닌 fragment를 구분한다.(예: SGML TAG, 숫자, 사람 이름 등)

3) 복합 자종의 표제어, 숙어, 정형 표현, 반드시 띄어 쓰는 특정 어휘(예:を) 등을 분리한다.

4) 각 fragment의 형태소 분석 방법을 제시한다.

입력 문장이 "Prologのプログラミングをはじめて學ぶ人には, The Art of Prolog

(L.Sterling E.Sharpiro 共著)の一讀を勸めておこう." 인 경우 전처리기의 처리 결과는 다음과 같다.

```
(   ;; Sent 1
  (("Prolog" ATM)("の")("プログラミング")("を" ATM)("はじめて")("學ぶ")("人には")(",")("The" ATM)
   ("Art" ATM)("of" ATM)("Prolog" ATM)("(" ATM)("L.Sterling" HUM)("E.Sharpiro" HUM)("共著")(")" ATM)
   ("の")("一讀")("を" ATM")("勸めておこう")("." ATM)) )
```

그림 3-2. 전처리기의 출력 예

전처리기가 제시하는 형태소 분석 정보의 의미는 다음 표와 같다. 제시하는 정보가 없는 것은 CYK 알고리즘에 따라 분석한다.

표 3-1. 전처리기가 제시하는 분석 유형

정보	의미
ATM	사전을 탐색하여 없으면 미등록어로 처리. 예: 복합 자종의 표제어, 정형 표현, 숙어
NUM	사전 탐색없이 '(명사 수사)'로 처리. 예: "1995", "-123,450,000.1234"
HUM	사전 탐색없이 '(명사 인명)'으로 처리. 예: "John F. Kenney"
TAG	Tag이므로 사전 탐색없이 무시. 예: "</paragraph>"
	이 이외의 경우에는 CYK알고리즘에 따라 분석. 예: "勸めておこう"

전처리기에서 사용하는 자종 정보는 히라가나, 가다가나, 한자, 영문자, 숫자, 기호이다. 장음 기호(0xA1BC:ー)는 가다가나 또는 히라가나로 취급하고 한자 반복기호(0xA1B8:**全**, 0xA1B9:**々**)와 한자zero를 의미하는 기호(0xA1B:〇)는 한자로 취급한다.

전처리기에서 하나의 단위로 처리하는 경우는 표 3-2와 같다. 이 이외의 경우에는 분리하여 처리한다. 일본어에서는 한자 다음에 히라가나가 오는 표제어의 비율이 약 7.8%(표 1-1)이고 동사, 형용사의 상당 부분이 이러한 유형이므로 이들은 하나의 fragment로 처리하는 것이 유리하다.

표 3-2 자종에 따른 분리(일부)

Unit	보기
자종i[자종i]*	日本, アメリカ, いい, 360, Korea
가다가나[가다가나|장음 부호]*	アメリカ, バベール,
한자[한자|한자 반복기호|〇]*	意氣揚々 , 各々, 共著 , 一九九〇年
한자[히라가나]*	韓國で, 遠い
[영문자|숫자]*	486DX2, Pentium90
[+|-]숫자([,]숫자)*[.]숫자*	+123,154.250
:	:

주) '*':0또는 1회 이상 반복, '|':선택적임을 의미

전처리기는 GNU의 scanner 생성기인 flex V2.5.1을 이용하여 구현되었으며 8,192개 이상의 specification rule을 처리할 수 있도록 flex source의 일부를 수정하였다. 전처리기

의 rule section에는 JUMAN V1.0[松本裕治93]의 사전에서 추출한 복합 자종의 표제어 8,475를 포함하여, 일한 기계번역에서 한 단위로 처리하는 것이 편리한 정형 표현 561개[森田良行92], 인명, 수 표현, 자종에 의한 분리 규칙 등이 specification rule로 기술되어 있다. flex를 이용함으로써 다양한 정규 표현을 이용하여 어떤 종류의 입력 형태에 대해서도 전처리기를 쉽게 적응시킬 수 있고, 전처리 과정의 고속화를 이룰 수 있다. 실제로 SDT-400 workstation(sun4/75Mhz)에서 254KB의 문서(2,379문장)를 처리하는데는 약 6초가 소요된다. 그러나, 복합 자종의 표제어가 새로 사전에 등록되는 경우, 이를 전처리기의 rule section에 추가하여야 하는 단점이 있다.

3.3 사전

사전은 JUMAN v1.0에서 사용한 EDR 사전을 기반으로 작성되었다. 일한 기계번역을 위하여 품사 체계를 정비하고 일본어 용언 처리를 위하여 용언 표제어를 수정하였다. 현재, 사전에는 고유명사를 제외한 84,458개의 표제어가 수록되어 있다. 고유명사는 별도로 관리하며 36,925개의 표제어가 수록되어 있다.

```
(あくる日 ((POS 명사 추상명사) (YOMI あくるひ)))
(あく促 ((POS 명사 사변명사) (YOMI あくせく)))
(あぐね ((POS 동사 모음동사) (YOMI あぐねる)))
(あけすけ ((POS 형용동사) (YOMI あけすけだ)))
(あげつら ((POS 동사 자음동사와행) (YOMI あげつらう)))
(あざと ((POS 형용사) (YOMI あざとい)))
(あざ笑 ((POS 동사 자음동사와행) (YOMI あざわらう)))
(あしら ((POS 동사 자음동사와행) (YOMI あしらう)))
(あしらい ((POS 명사 보통명사) (YOMI あしらい)))
(あすこ ((POS 명사 보통명사) (YOMI あすこ)))
(あずまコード ((POS 명사 보통명사) (YOMI あずまこーど)))
```

그림 3-3. 형태소 분석 사전의 일부

표제어는 일본어 EUC코드를 사용하며 그 외 품사 및 음독 정보는 KS C-5601 완성형 코드를 사용한다. 사전은 GNU의 gdbm library를 이용하여 gdbm화 시켜 사용하며 별도의 interface를 두어 실시간으로 사전 엔트리의 추가, 삭제, 변경이 가능하도록 하였다.

3.4 접속 정보표

접속 정보표는 형태소간의 접속 관계를 검증하는데 이용된다. 접속정보표는 숫자, symbol 등을 이용하여 기술하는 것이 일반적이나 이는 가독성이 떨어져 접속 정보표 수정 및 유지를 어렵게 한다. 본 시스템에서는 JUMAN에서와 같이 품사를 직접 기술하도록 하며, 접속 체크에 소요되는 시간을 최소화하기 위하여 이를 matrix로 변환시켜 분석시 이용한다. 품사는 최대 3단계까지 하위범주화되어 총 253개로 분류되어 있으며 접속정보표는

2,048라인(34KByte)이며 실험을 통해 계속 보완하고 있다.

```
접속정보표 ::= 접속정보+
접속정보 ::= '(' 좌범주리스트  우범주리스트 ')'
좌범주리스트 ::= '(' 좌범주+ ')'
우범주리스트 ::= '(' 우범주+ ')'
좌범주 ::= '(' 품사+ ')'
우범주 ::= '(' 품사+ 문자열? ')'
품사 ::= 한글+ | Wild_Character
Wild_Character ::= '*'
문자열 ::= "일본어+"

주) '+':1회 이상 반복, '?':optional, '|':선택적임을 의미
```

그림 3-4 접속 정보 기술 syntax

```
(((동사 자음동사か행) (부속사 * 자음동사か행))
 ((동사어미 동사기본형 "く") (동사어미 동사미연형 "か") (동사어미 동사의지형 "こう")
  (동사어미 동사명령형 "け") (동사어미 동사기본조건형 "けば") (동사어미 동사기본연용형 "き")
  (동사어미 동사타형 "いた") (동사어미 동사타계추량형 "いたろう") (동사어미 동사타계조건형 "いたら")
  (동사어미 동사타계연용테형 "いて") (동사어미 동사타계연용타리형 "いたり") (부속사 자음동사か행)))
                         :
(((명사 사변명사))
 ((명사) (부사) (형용사) (특수) (문말) (접속사) (동사 사변동사) (부속사 명사) (판정사)))
```

그림 3-5 접속 정보표(일부)

```
char *CONNECTIVITY_TABLE[] ={
  "0011111111111111111111111111111111111111111111111111111111111110000000000000000000000000000000000000
0000000000000000000000000000000000000000000000000000000000000111111000000000000000000000000000000
0000000000000000000000000000000000000000000000000000000000000111111010000000000000000000000000000100
0000000000000000000000000000100000000000",
                         :
  "0101111111111111111111111111111111111111111111111111111111111110000000000000000000000000000000000000
0000000000000000000000000000000000000000000000000000000000000111111000000000000000000000000000000
0000000000000000000000000000000000000000000000000000000000011101111101111000000000000000000000000100
0000000000000000000000000000100000000000"
  };
```

그림 3-6 matrix로 변환된 접속 정보(일부)

4. 분석 알고리즘 및 수행 예

4.1 알고리즘

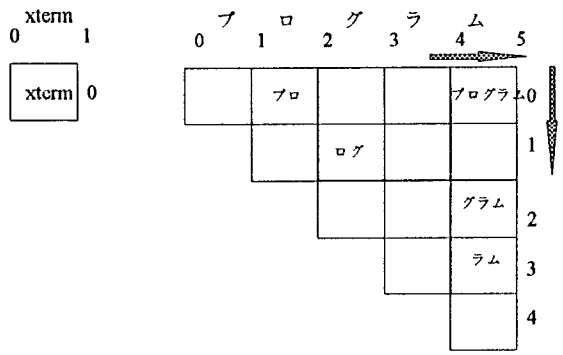
Algorithm DaHae(j_paragraph_prep_out);

```
step 1: 전처리기의 분석 결과 중 한 문장의 fragment리스트(frag_list)를 가져온다;
step 2: chart, graph, graph의 entry테이블을 초기화한다;
step 3: 앞 fragment의 terminal_entry_stack을 '문두'로 초기화;
step 4: frag_list에서 한 fragment를 가져온다;
step 5: switch (fragment의 분석 정보) {
        case ATM:            dic_info = dic_look_up(fragment);
                             fragment가 미등록어이면  dic_info ='(명사 보통명사)';
```

```
                    simple_analysis(dic_info, fragment);
    case NUM:       dic_info ='(명사 수사)';
                    simple_analysis(dic_info, fragment);
    case HUM:       dic_info ='(명사 인명)';
                    simple_analysis(dic_info, fragment);
                    :
                    :
    default:        CYK_analysis(fragment);
    )
step 6: frag_list가 null일 때까지 step 4,5 반복;
step 7: 가능한 모든 분석 path 추출;
step 8: 최적분석 path 선택;
step 9: 최적분석 path 출력;
step 10: 수행정보(사전 탐색 시도 횟수 및 성공 횟수, 접속 체크 시도 횟수 및 성공 횟수) 출력;
step 11:다음 문장이 없을 때까지 step 1~10 반복;
```

그림 4-1. 형태소 분석 알고리즘

본 시스템은 기본적으로 chart 자료 구조를 이용한다. 전처리기가 제시한 분석 정보가 ATM인 경우에는 fragment전체를 1회 사전 탐색하여 얻은 정보를, HUM, NUM, TAG 등인 경우에는 사전 탐색 없이 미리 정의된 품사 정보를 chart의 한 cell에 수록하여 분석을 시도한다. 그 밖의 경우(CYK식인 경우)에는 fragment의 음절 길이(n)에 따라 n*(n+1)/2 크기의 삼각 테이블(T)를 구성하고 fragment의 i(i>=0)번째 문자부터 j번째 문자까지의 부분문자열을 사전 탐색하여 테이블의 원소 T(i,j)를 채운다.



a) ATM, NUM, HUM인 경우　　b)CYK식 분석인 경우

그림 4-2. 전처리기의 분석 정보에 따른 chart의 구성

형태소 분석기는 chart의 인접한 형태소간의 접속 체크를 시도하여 접속 가능하면 이를 DAG(Direct Acyclic Graph)에 수록한다. fragment와 fragment간의 접속 체크를 위하여 접속 가능한 경우, 현재 처리 중인 fragment의 마지막 문자로 끝나는 부분문자열의 품사 정보를 Terminal Entry Stack에 저장하고, 다음 fragment 처리 시에 fragment의 첫 문자로 시작하는 엔트리(i=0, j>1)처리시에 이들과 접속 체크를 시도하여 접속 가능하면 DAG에 수록한다. 예를 들어, 그림 4-2 b)에서 'プログラム', 'プロ'+'グラム', 'ログ'+'ラム'가 접속 가능하다면 Terminal Entry Stack에는 'プログラム','グラム','ラム'가 저장된다. 문장

의 맨 처음 fragment는 가상의 fragment '문두'와 접속 체크하도록 하여 형식형태소로 시작하는 해석 결과를 배제시키고 문장의 맨 끝에는 가상의 fragment인 '문말'을 둔다.

한 문장에 대한 분석이 끝나면 구성된 그래프에서, '문말'까지 이어지지 못하는 arc들을 제거하여 가능한 모든 분석 path를 얻는다.

4.2 수행 예

입력 문장이 "Prologのプログラミングをはじめて學ぶ人には, The Art of Prolog (L.Sterling E.Sharpiro 共著)の一讀を勸めておこう." 인 경우 실제로 분석되는 과정은 그림 4-3와 같다.

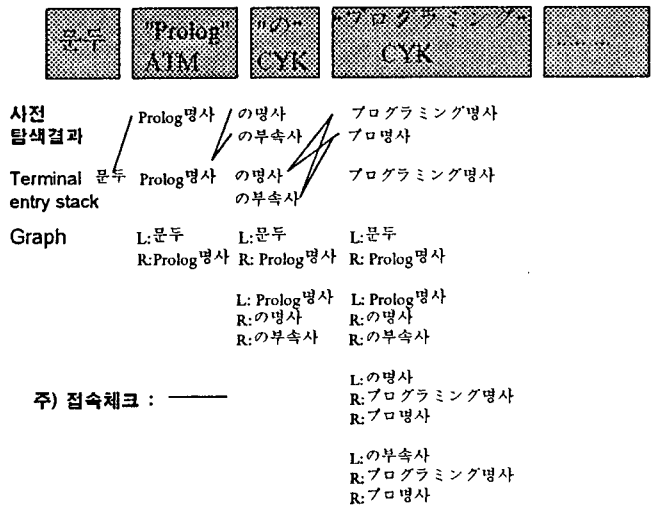


그림 4-3 분석 과정의 예

전처리기에 의해 입력 문장은 그림 3-2에서와 같이 22개의 fragment로 분리된다. 맨 처음 fragment인 "Prolog"는 전처리기가 제시한 정보(ATM)에 따라 사전을 1회 탐색하여 이를 chart에 수록한다. "Prolog"가 맨 처음 fragment이므로 '문두'와 접속 체크를 하고 접속 가능하므로 이를 그래프에 등록한다. 또한, "Prolog"가 fragment의 마지막 문자를 포함하므로 이를 Terminal Entry Stack에 등록하고 다음 fragment "の"의 분석 시 접속 체크를 한다.

5. 실험

DaHac는 sun 호환 국산 workstation인 SDT-400(sun os 4.1.2, X11R5)에서 GNU의 컴파일러 제작 도구인 flex와 bison을 이용하여 c언어로 구현되었다. 본 분석기의 성능 평가를

위해 [長尾 眞]의 2,261문장을 가지고 실험을 하였고, 분석 결과에 따라 사전 및 접속 정보표를 보완하는 작업이 현재 진행 중이다.

실제로 분석된 결과의 예는 그림 5-1와 같다.

입력 문장

一つは濃淡が急激に變化しているところを取り出すもので, これは劃像の濃淡値を數學的に微分することによってえられる.

전처리 결과

((("一つは") ("濃淡が") ("急激に") ("變化しているところ") ("を" ATM) ("取り出" ATM) ("すもので") (", " ATM) ("これは") ("劃像の") ("濃淡値") ("を" ATM) ("數學的に") ("微分することによってえられる") (". " ATM)))

최적해석 결과

```
((一つ ((POS 명사 보통명사) (YOMI ひとつ)))
(濃淡 ((POS 명사 보통명사) (YOMI のうたん)))
(急激 ((POS 형용동사) (YOMI きゅうげきだ)))
(變化 ((POS 명사 사변명사) (YOMI へんか)))
(い ((POS 부속사 동사타계연용데형 모음동사)(YOMI いる)))
(ところ ((POS 명사 보통명사) (YOMI ところが)))
(取り出 ((POS 동사 자음동사사행) (YOMI とりだす)))
(もの ((POS 명사 형식명사) (YOMI もの)))
( ` ((POS 특수 독점) (YOMI ` )))
(は ((POS 부속사 명사 조사)(YOMI は)))
(の ((POS 부속사 명사 조사)(YOMI の)))
(値 ((POS 명사 사변명사) (YOMI あたい)))
(數學的 ((POS 명사 보통명사) (YOMI すうがくてき)))
(微分 ((POS 명사 사변명사) (YOMI びぶん)))
(こと ((POS 명사 형식명사) (YOMI こと)))
(よ ((POS 동사 자음동사와행) (YOMI よう)))
(え ((POS 동사 자음동사라행) (YOMI える)))
(れ ((POS 부속사 동사미연형 모음동사)(YOMI れ)))
( ° ((POS 특수 구점) (YOMI ° ))) )
(は ((POS 부속사 명사 조사)(YOMI は)))
(が ((POS 부속사 명사 조사)(YOMI が)))
(に ((POS 형용동사어미 형용동사다열기본연용형)(YOMI に)))
(して ((POS 동사 사변동사 동사타계연용데형)(YOMI して)))
(る ((POS 동사어미 동사기본형)(YOMI る)))
(を ((POS 부속사 명사 조사)(YOMI を)))
(す ((POS 동사어미 동사기본형)(YOMI す)))
(で ((POS 판정사 판정사다열타계연용데형)(YOMI で)))
(これ ((POS 명사 보통명사) (YOMI これ)))
(劃像 ((POS 명사 보통명사) (YOMI がぞう)))
(濃淡 ((POS 명사 보통명사) (YOMI のうたん)))
(を ((POS 부속사 명사 조사)(YOMI を)))
(に ((POS 부속사 명사 조사)(YOMI に)))
(する ((POS 동사 사변동사 동사기본형)(YOMI する)))
(に ((POS 부속사 명사 조사)(YOMI に)))
(って ((POS 동사어미 동사타계연용데형)(YOMI って)))
(ら ((POS 동사어미 동사미연형)(YOMI ら)))
(る ((POS 동사어미 동사기본형)(YOMI る)))
```

Dic Access : 211 Dic Found : 70 Match Trial : 986 Match Success : 333

그림 5-1. 형태소 분석의 예

한 문장에 대한 분석은 어휘적, 품사적 중의성으로 인하여 여러 개가 될 수 있다. 이들 중 최적의 분석 결과를 결정하는 방법에는 최장일치법, 2문절 최장일치법, 문절수최소법, 최대평가치법 등 다양한 방법들이 제시되어 왔다[김은자94][시스템89]. 그러나, 형태소 분석 단계에서 최적 분석 선택에는 한계가 있으므로 형태소 분석 단계에서 처리할 수 없는 중의성은 구문 분석이나 의미 분석 단계로 넘기는 것이 바람직하다.

본 시스템에서는 가능한 모든 분석 path를 graph로 출력하지만 변환기와의 interface를 위하여 가능한 분석 중 최장일치법을 이용하여 하나의 분석 path를 최종 결과로 출력한다. 현재, 통계적 모델을 이용하여 최적 path를 선택하기 위한 corpus 구축 및 통계 정보를 추

출하는 작업을 진행 중에 있다. 이 작업이 완성되면 통계 정보를 이용하여 평가치가 큰 n 개를 최종 결과로 출력시킬 예정이다.

6. 결론

본 논문에서는 형태소 분석의 전단계로 전처리기를 두어 입력 문장을 자종의 천이에 따라 여러개의 fragment로 분리하고 복합 자종 표제어, 인명, 정형 표현 등을 인식하여 이들의 형태소 분석 방법을 제시하고, 형태소 분석기는 이에 따라 분석을 하는 시스템을 제안하였다. 이 방법은 새로운 형태의 입력 표현(예: HTML문서에서의 처리)에 대한 형태소 분석기의 적응성(flexibility)을 높이고, 불필요한 사전 탐색과 분석 과정을 줄여 시스템의 성능을 향상시킨다. 예를 들어, "なければならない"와 같은 정형 표현을 하나의 번역 단위로 분석하면 "な+ければ+な+ら+な+い"와 같이 각 구성 형태소를 분리한 후, 이들의 접속 관계를 조사하는 것보다 분석의 노력을 절감하고 한국어 대역어를 일의적으로 결정할 수 있어 효과적이다. 본 시스템에서는 이러한 정형 표현들을 전처리기가 하나의 fragment로 인식해 내고 해석 방법을 제어함으로써 불필요한 사전 탐색 횟수과 접속 체크 횟수를 크게 줄인다.

입력 문장이 히라가나로만 작성된 경우, 전체 문장이 하나의 fragment가 되고 이를 CYK 알고리즘에 따라 분석하게 되면 불필요한 사전 탐색을 많이 수행하고 모호성도 많이 발생하게 된다. 그러나, 국민학교 교과서나 외국인을 위한 교재 이외에는 일본어 문서 작성시 가능한 한 한자를 섞어 쓰는 것이 원칙이므로 본 논문에서 제안된 방법은 타당성을 갖는다고 할 수 있다.

현재의 시스템은 prototype 시스템으로 아직 미등록어 처리 기능이 미약하여 입력 문장 중 미등록어가 존재하면 분석이 실패할 가능성이 크므로 미등록어 처리 기능을 보강하여야 하며, 가능한 모든 분석 path를 평가하여 평가치가 큰 일부의 path를 출력하는 연구가 진행되고 있다.

참고 문헌

[강석훈95] 강석훈, 최병욱, "일한 번역시스템을 위한 일본어 분석기 설계," 전자공학회논문지, 제3권 B편, 제1호, pp.136-146, 1995.

[김은자94] 김은자, 연어 패턴에 기반한 일-한 기계 번역 시스템, 포항공대 전자계산학과 석사학위논문, 1994

[시스템89]시스템공학센터, 한일.일한 자동번역시스템 개발에 관한 연구(III),1989

[HIROSHI80] Hiroshi Nakano, et al., "An Automatic Processing of the Natural Language in the Word Count System," Proceeding of the 8th International Conference on Computational Linguistics, pp. 338-345, 1980

[Lunde92] Ken R. Lunde, *Electronic Handling of Japanese Text*, Adobe Systems Inc., 1992.

[MARUYAMA88] N. Maruyama, et al., "A Japanese sentence Analyzer," IBM J. Res. Develop. Vol. 32, No. 2, pp. 238-250, 1988

[森田良行92] 森田良行 외 1인, 日本語 表現 文型 -用例中心.複合辭の意味と用法,アルク, 1992

[松本裕治93] 松本裕治 외4인, 日本語形態素解析システムJUMAN version 1.0 使用說明書, 1993, 京都大學工學部 長尾연구실

[松本裕治94] 松本裕治 외4인, 日本語形態素解析システムJUMAN version 2.0 使用說明書, 1994, 京都大學工學部 長尾연구실

[長尾 眞] 長尾 眞, 人工知能と人間