DOI: 10.3745/KIPSTB.2010.17B.3.255

한국어 자연어 요구문서에서 구문 구조 기반의 조응어 처리 시스템

박 기 선[†] · 안 동 언[††] · 이 용 석[†††]

요 약

시스템 개발에 있어서 요구문서(requirements document)를 생성하고 정형 명세를 작성하는 것은 요구 분석 전문가와 명세 전문가에 의해 수행되고 있다. 만약 요구문서 생성과 정형 명세 작성 과정을 자동화 한다면 시스템 개발 비용 및 기간을 단축할 수 있고, 또한 전문가 사이의 잘못된 이해로 인한 오류를 줄일 수 있다. 대명사는 인칭대명사와 지시대명사로 분류될 수 있다. 일반적으로 요구문서의 특성상 인칭대명사는 사용되지 않기 때문에 본 논문은 지시대명사의 지시어 결정에 초점을 두고 있다. 지시대명사를 포함하는 요구문서에서 자연어처리 기법을 통해 정형화된 요구사항을 자동으로 추출하기 위해서는 대명사의 지시어 결정이 매우 중요하다. 본 연구의 최종 목표는 자연어 처리 기법을 통하여 자연어 요구문서로부터 시스템 개발에 필요한 정형 명세를 자동으로 생성하는데 있다. 이를 위해 본 논문은 선행연구를 기반으로 한국어로 기술된 자연어 요구문서에서 대명사에 대한 지시어를 결정하는 조응어 해소(anaphora resolution) 시스템을 제안한다. 본 시스템의 개발을 위해 조응어 해소를 위한 경험 규칙을 정의하고, 이를 통해 10개의 요구문서에 대해 실험한 결과 평균 재현율 92.45%, 정확률 69.68%의 성능을 보였다.

키워드 : 조응어 해결, 선행사 결정, 자연어처리, 요구사항 추출

Anaphora Resolution System for Natural Language Requirements Document in Korean based on Syntactic Structure

Ki-Seon Park[†] · Dong-Un An[††] · Yong-Seok Lee[†††]

ABSTRACT

When a system is developed, requirements document is generated by requirement analysts and then translated to formal specifications by specifiers. If a formal specification can be generated automatically from a natural language requirements document, system development cost and system fault from experts' misunderstanding will be decreased. A pronoun can be classified in personal and demonstrative pronoun. In the characteristics of requirements document, the personal pronouns are almost not occurred, so we focused on the decision of antecedent for a demonstrative pronoun. For the higher accuracy in analysis of requirements document automatically, finding antecedent of demonstrative pronoun is very important for elicitation of formal requirements automatically from natural language requirements document via natural language processing. The final goal of this research is to automatically generate formal specifications from natural language requirements document. For this, this paper, based on previous research [3], proposes an anaphora resolution system to decide antecedent of pronoun using natural language processing from natural language requirements document in Korean. This paper proposes heuristic rules for the system implementation. By experiments, we got 92.45%, 69.98% as recall and precision respectively with ten requirements documents.

Keywords : Anaphora Resolution, Antecedent Decision, Natural Language Processing, Requirements Elicitation

1. 서 론

자연어로 기술된 문서에 있어서 문서 작성의 경제성을 고려하여 용어 반복을 대신하는 대명사의 사용이 빈번하다.

일반적인 대용 현상은 말의 경제성에 따라 문장의 구성 요소인 선행어가 먼저 등장하고 이후에 대용어(조응어)가 대치되어 사용되는 순행 대응을 뜻한다[1]. 이러한 대용 현상의 처리는 기계번역, 질의응답, 문서분류 및 요약, 정보검색 등 자연어 처리 시스템에서 필수적으로 요구된다. 그러나 요구문서에 있어서는 그 특성상, 인칭대명사는 거의 사용되지 않는다. 따라서 본 연구는 자연어로 기술된 요구문

† 준 회 원 : 전북대학교 컴퓨터공학과 박사수료
†† 종신회원 : 전북대학교 IT정보공학부 교수
††† 정 회 원 : 전북대학교 컴퓨터공학과 교수
논문접수 : 2010년 1월 18일
수 정 일 : 1차 2010년 2월 24일, 2차 2010년 3월 9일
심사완료 : 2010년 3월 10일

서에 있어서 지시대명사의 지시어 결정 문제에 초점을 두고 있다. 'Anaphor'는 조응어, 조응사, 대용어, 대용사 등으로 불리나 본 논문에서는 연구[2]의 표현에 따라 조응어라 부르기로 한다. 조응어는 하나의 단어 단위 표현뿐만 아니라 구 단위 표현도 지시 대상이 된다는 의미를 포함한다[2].

본 연구의 목적은 한국어 자연어로 기술된 요구문서로부터 요구 명세를 자동으로 생성하기 위해 지시대명사를 지시어로 대치하는 시스템을 구현하는 것이다. 이를 위해, 연구[3]에서 규칙을 기술하였고, 이를 기반으로 규칙을 보완하여 시스템을 구현하였다.

일반적으로 한국어에 있어서 대명사의 특징은 다음과 같다[4].

1) 대명사는 단독으로 주어가 될 수 있다.
   예) 이것은 무엇을 위한 물건이냐?
2) 대명사는 조사가 붙어 격표시가 이루어진다.
   예) 이것이(주격), 이것을(목적격), 이것의(관형격)
3) 대명사는 관형사의 수식을 받을 수 없다.
   예) 나는 새 그것을 사고 싶었다. (X)
   나는 새 책을 사고 싶었다. (O)
4) 대명사는 용언의 관형사형의 수식은 받을 수 있다.
   예) 아름다운 이것도 결국은 쓸모가 없다.

자연어 요구문서로부터 정형화된 요구사항을 자동으로 정확히 추출하기 위해서는 대명사의 특징을 고려하여 조응어를 결정하는 것이 매우 중요하다.

(그림 1)은 [5]에 기술된 자연어 요구문서에 대명사를 추가한 예의 일부를 보이고 있다.

대명사에 대한 조응어는 형태소 분석과 구문 분석 단계에서 결정 될 수도 있다. 그러나 본 연구에서는 분석단계의 부하를 줄이고, 규칙을 유연하게 수정 및 추가 할 수 있도록 하기 위하여 후처리 단계에서 조응어를 처리하는 시스템을 제안한다.

가) 일반적 요구사항
이 시스템은 커피, 콜라, 코코아를 종이컵으로 판매하는 자판기이다.
자판기는 이것들을 모두 백 원에 판매한다.
이것이 팔 수 있는 음료수는 각각 10잔씩 이다.
이것은 종이컵을 20개 가지고 있을 수 있다.

나) 전기관련 요구사항
이것은 전원이 켜지기 전에는 동작하지 않는다.
이것은 전원이 차단되면 즉시 모든 동작을 멈춘다.

다) 동전 관련 요구사항
이것이 받을 수 있는 동전은 500, 100, 50, 10원이다.
이것의 반환레버를 작동해야지 거스름돈을 반환한다.
아래의 상태에서 돈을 집어넣으면 바로 반환된다.
(1) "판매중"에 불이 안 들어와 있을 경우
(2) "전원"이 들어와 있지 않을 경우

(그림 1) 대명사를 포함한 자연어 요구문서의 예

본 논문의 구성은 다음과 같다. 2장에서는 관련 연구에 대해 기술하고, 3장에서는 시스템의 구조를 보이고, 4장에서는 자연어 요구문서에서 지시대명사의 처리 규칙 및 방법에 대해 설명하고, 5장에서는 실험 및 평가에 대해 논한다. 그리고 마지막으로 결론 및 향후 연구로 끝을 맺는다.

2. 관련 연구

대용 현상 해소에 관한 연구는 영어, 스페인어, 한국어를 비롯한 많은 언어들에서 이루어져 왔다. 그러나 많은 연구들이 인칭대명사의 처리에 초점을 두고 있다.

특히 정보검색 시스템에서는 특정 인물의 이름과 같은 고유 명사가 '그', '그녀'와 같은 인칭 대명사로 표현되어 있으면 검색하고자 하는 용어 가중치가 다르게 계산되어 원하는 검색 결과를 얻을 수 없다. 이러한 이유로 많은 연구들이 인칭 대명사의 처리에 초점을 맞추어 진행되었다[1, 6, 7].

연구[1]은 문서요약을 위해 실마리어(단서 단어: Seed word)를 중심으로 기술된 휴리스틱 룰과 센터링 이론(Centering theory)을 적용하여 개체명에 대한 대용 해소 시스템을 제안하였다.

연구[6]은 정확한 분석을 위해 대용 해소 범위를 3인칭 대명사로 제한하고, 대명사의 격에 따라 7개의 경험 규칙을 제안하였다. 또한, 연구[7-9]는 텍스트를 구조화 하거나, 지시어(대명사)를 생성하거나, 대용과 생략을 해결하는데 센터링 이론을 적용하였다.

연구[7]은 일반 문장에서 먼저 영-대명사(Zero-pronouns)를 찾고, 그 위치에 대명사를 삽입하여 대명사의 품사와 문장의 범위를 반영하여 지시어를 결정하고 있다.

또한, 지시대명사에는 실마리어가 존재하지 않을 뿐 아니라, 한국어는 조사에 의해 격표시가 이루어지고, 어순이 매우 자유로운 특성이 있다. 이러한 이유로 연구[10]은 한국어에서 센터링 이론으로 대용 문제를 해결하기 위해서는 고정된 Cf(Forward-looking-centers)[1] 목록의 우선순위와 센터의 전이 유형간의 우선순위를 신뢰할만한 기준으로 삼기에는 한계가 있음을 보이고 있다. 따라서 한국어에 센터링 이론을 적용하기 위해서는 통계를 기반으로 확률을 적용하여 Cf 목록과 센터 전이 유형을 동적으로 적용하고, 지식 기반의 경험 규칙을 추가로 기술함이 타당할 것이다. 그러나 이를 위해서는 많은 비용과 시간이 소요된다. 또한, 연구[11]은 센터링 이론만으로 영-대명사의 선행사를 결정할 수 있는 장치를 구성하는 것은 많은 어려움이 있음을 보이고 있다. 연구[11]을 토대로 영-대명사를 대명사로 대치할경우도 같은 결과를 보일 것이다. 따라서 본 논문에서는 문장의 모호함이 적은 요구문서의 특성을 고려하여 경험 규칙들을

[1] Cf(Forward-looking-centers) 목록은 발화에 나타난 담화 요소의 집합으로 문법적 역할에 따라 우선순위가 부여된다.

정의하고, 이를 통해 대용 현상을 해소하고자 한다.

본 논문은 지시대명사의 격에 따라 9개의 경험 규칙을 기술하고, 한국어의 특징을 고려하여 세부 조건에 따라 지시대명사에 대한 지시어를 결정하는 시스템을 제안한다. 9개의 경험 규칙을 기술하기 위하여, 연구[5, 12, 13]을 참고로 요구문서에 다양한 패턴의 지시대명사를 추가하여 작성하였다.

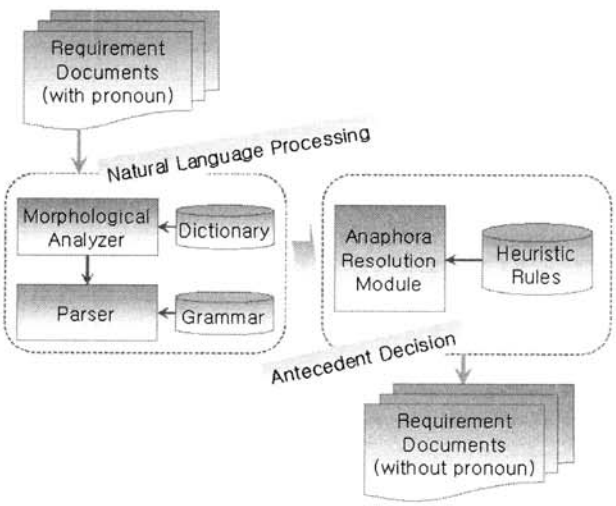
3. 시스템 구조

본 연구에서 제안하는 자연어 요구문서 조응어 해소 시스템은 크게 자연어 처리부와 조응어 처리부로 구성된다. 자연어 처리부는 형태소 분석기와 구문 분석기로 구성되는데, 이는 연구[14]의 분석기를 사용하였다.

본 연구에 사용된 형태소 분석기는 구문 형태소를 이용하여 형태론적 모호성을 줄여주어 구문 분석의 효율을 높여주는 특징이 있다. 구문 분석기는 구구조 문법에 기반한 조건 단일화 방법을 이용하여, 구구조 규칙의 간결함과 구구조 의존적 언어의 특성을 조건 단일화 제약을 통해 문장을 분석하는 특징이 있다.

(그림 2)는 본 시스템의 구조를 보이고 있다.

지시대명사를 포함하는 자연어 요구문서는 자연어 처리부로 입력되어 형태소 분석과 구문분석 단계를 거쳐 조응어 처리부의 입력으로 제공된다. 조응어 처리부는 구문 분석 결과 정보 및 28개의 지시대명사 등을 포함한 정보에 기반한 경험 규칙에 의해 지시대명사에 대한 지시어를 탐색하여 지시대명사를 이로 대체하게 된다.



(그림 2) 자연어 요구문서 조응어 해소 시스템 구조

4. 조응어 처리

이 장에서는 본 연구에서 처리할 대명사의 범위와 조응어에 대한 제약 조건 및 처리 규칙에 대해 설명한다.

4.1 대명사의 범위

본 연구에서 다루는 대명사의 범위는 다음과 같다.

가. 지시대명사

나. 의미 분석이 필요하지 않은 경우

- 의미 분석이 필요한 예

"그가 바쁠 지라도 시스템은 사용자의 명령을 처리한다."

다. 문맥 분석이 필요하지 않은 경우

- 문맥 분석이 필요한 예

"시스템은 그가 한국에 있을 때 그의 주인을 도왔다."

라. 지시어가 여러 문장에 걸쳐 등장하지 않는 경우

- 여러 문장에 걸쳐 조응어가 등장하는 예

"시스템은 15분 이내에 오븐에서 치킨을 요리한다. 튀김은 10분 이내에 레인지에서 튀겨야 한다. 이들 모두는 연속적으로 이루어져야 한다."

마. 주어, 목적어, 부사어, 관형어, 지시관형사로 제한.

4.2 지시어 요건

본 연구에서 다루는 조응어의 요건은 다음과 같다.

가. 현재 문장의 바로 이전 문장으로 제한

나. 주격, 목적격 어절

- 동격에 의해 수식되면 조응어는 동격으로 확장
- 관형어의 경우 문장 성분 무관

다. 대명사와 수가 일치하는 명사

- 복수형 대명사의 경우 나열형을 모두 조응어로 선택

4.3 지시어 결정 규칙

이 절에서는 지시대명사의 지시어 결정을 위한 경험 규칙을 기술한다. 지시어 결정 규칙은 다음과 같은 기호를 정의하여 기술한다.

Fs(요구문서의 첫문장), Pr(이전문장), Cr(현재문장), P(문장성분), D(지시대명사), A(조응어), L(왼쪽어절), R(오른쪽어절), U(관계없음), Null(없음)

RULE(1) :

if L = substantive without Josa
then CrDA -> L

조사 없는 체언 뒤에 바로 지시대명사가 나타날 경우 앞선 체언을 지시어로 선택하고, 체언의 나열 뒤에 바로 지시대명사가 나타날 경우 나열된 체언들을 지시어로 선택한다.

예) "커피 이것은 백 원이다."

∴ '이것'의 지시어는 '커피'

예) "자판기는 커피, 콜라, 코코아 이것들을 모두 백 원에 판매한다."

∴ '이것들'의 지시어는 '커피 콜라 코코아'

RULE(2) :

if !Fs &
CrDP = PrDP &

CrD = PrP &
CrDP ≠ Obj
then CrDA -> PrDA

이전 문장에서 현재 지시대명사와 문자열이 정확히 일치되는 것을 찾아 지시어로 선택한다. 이때, 일치되는 문자열이 지시어를 갖고 있으면 다른 규칙에 관계없이 이로 확장한다. 현재 문장의 지시대명사가 목적격이고, 확장된 조응어가 현재 문장에 동일 문장 성분으로 존재하면 RULE(2)를 생략하고 해당 규칙으로 처리한다.

예) "이것은 자판기이다."
"이것은 커피를 판매한다."
"이것은 백 원이다."
∴ 첫 번째 문장에서 '이것'의 지시어는 "자판기"
두 번째 문장에서 '이것'의 지시어는 '자판기'
세 번째 문장에서 '이것'의 지시어는 '자판기'
(* RULE(6.3)에 앞서 여기에서 먼저 처리)

예2) "이것은 자판기이다."
"이것은 커피를 판매한다."
"자판기는 이것을 백 원에 판매한다."
∴ 첫 번째 문장에서 '이것'의 지시어는 "자판기"
두 번째 문장에서 '이것'의 지시어는 '자판기'
세 번째 문장에서 '이것'의 지시어는 '커피'
(* RULE(7.1)에 의해 처리)

RULE(3) :
if CrDP = ADJ-MOD
then CrDA -> R

지시관형사일 경우 바로 뒤 어절을 지시어로 선택

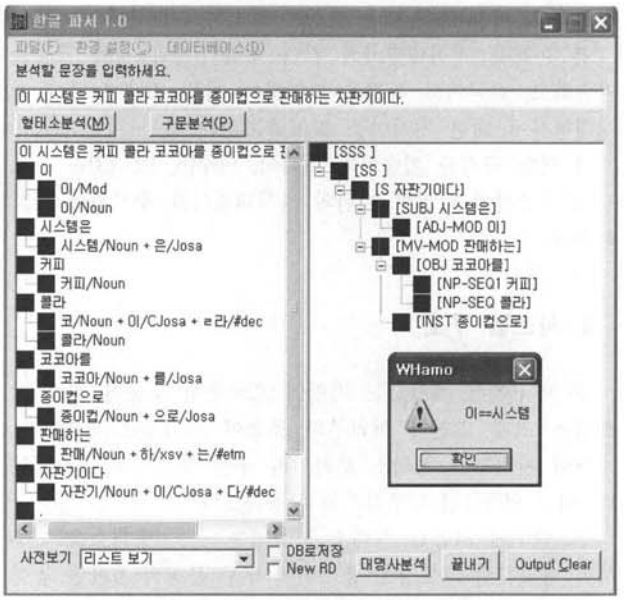
예) "이 시스템은 커피 콜라 코코아를 종이컵으로 판매하는 자판기이다."
∴ '이'의 지시어는 '시스템'

(그림 3)은 RULE(3)에 의한 조응어 처리 결과를 보인다.

RULE(4) :
if Fs &
CrDP = Subj &
CrP is S_predicate &
S = (complement | substantive with
predicative case josa '이다')
then CrDA -> Root(S)

지시대명사가 요구문서의 첫 문장에 주어로 등장할 경우 현재 문장의 보어 또는 체언에 서술격 조사 '이다'가 붙은 술부의 어근을 지시어로 선택한다. 이때 용언의 명사형은 형태소 분석 결과를 통해 배제된다.

예) "이것은 커피 콜라 코코아를 종이컵으로 판매하는 자



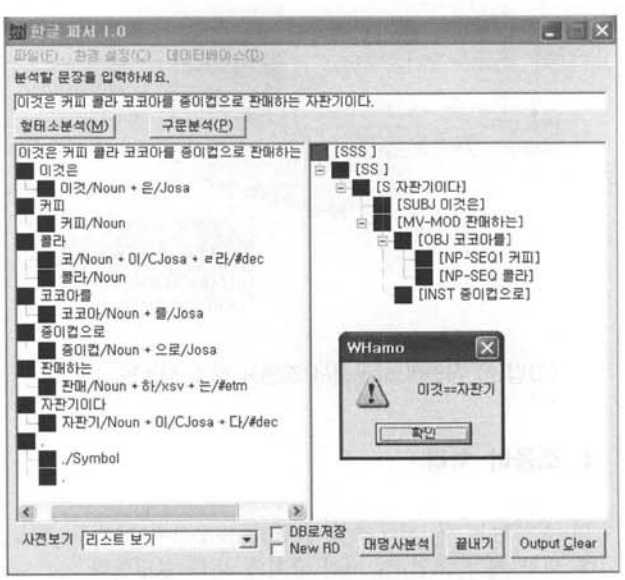
(그림 3) RULE(3)에 의한 조응어 처리 결과

판기이다."
∴ '이것'의 지시어는 '자판기'

(그림 4)는 RULE(4)에 의한 조응어 처리 결과를 보인다.

(그림 4)에서 보이는 바와 같이 '자판기이다'는 구문분석 결과 술어로 분석되므로 형태소 분석 결과를 통해 어근 '자판기'를 지시어로 선택하게 된다.

RULE(5) :
if Fs &
CrD = null &
CrP ≠ Subj
then CrD -> "이것은"
CrDP -> Subj



(그림 4) RULE(4)에 의한 조응어 처리 결과

CrDA -> apply(RULE(4))

요구문서 첫 문장에 주어가 없고, 첫 머리에 영-대명사가 등장할 경우는 영-대명사 자리에 "이것은"을 삽입 후 RULE(4)에 의해 지시어 선택.

예) "커피, 콜라, 코코아를 판매하는 자판기이다."

∴ 영-대명사의 지시어는 '자판기'

위의 예문은 RULE(5)에 의해 문장 "이것은 커피, 콜라, 코코아를 판매하는 자판기이다."로 변형된다. 이후, RULE(4)에 의해 '자판기'를 지시대명사 '이것'의 지시어로 선택하게 된다.

현재 문장에서 지시대명사가 주격일 경우 조건에 따라 지시어 결정한다.

RULE(6.1) :

```
if !Fs &
   CrDP = Subj &
   CrP ∋ Obj &
   PrP ∋ Subj
then PrP -> Subj
     CrDA -> Root(PrP)
```

현재 문장에 목적어가 있을 경우 이전 문장의 주어를 지시어로 선택한다.

예) "은행은 계좌의 목록을 유지한다."

"그것은 ID와 PIN을 검사한다."

∴ '그것'의 지시어는 '은행'

RULE(6.2) :

```
if !Fs &
   CrDP = Subj &
   CrP ∋ Obj &
   PrP ∋ Subj >= 2
then PrP -> the first Subj
     CrDA -> Root(PrP)
```

현재 문장에 목적어가 있고 이전 문장에 주어가 둘 이상일 경우 이전 문장의 주어 중 선행하는 주어를 지시어로 선택한다.

예) "자판기가 받을 수 있는 동전은 백 원이다."

"이것은 반화레버를 동작하면 거스름돈을 반환한다."

∴ '이것'의 지시어는 '자판기'

RULE(6.3) :

```
if !Fs &
   CrDP = Subj &
   CrP ∌ Obj &
   PrP ∋ Obj
then PrP -> the last Obj
```

CrDA -> Root(PrP)

현재 문장에 목적어가 없고 이전 문장에 목적어가 있을 경우 이전 문장의 목적어를 지시어로 선택한다.

예) "자판기는 커피, 콜라, 코코아를 판매한다."

"이것들은 모두 백 원이다."

∴ '이것들'의 지시어는 '커피, 콜라, 코코아'

RULE(6.4) :

```
if !Fs &
   CrDP = Subj &
   CrP ∌ Obj &
   PrP ∌ Obj &
   CrP ∋ Subj
then PrP -> the last Subj
     CrDA -> Root(PrP)
```

현재 문장에 목적어가 없고 이전 문장에 목적어 없을 경우 이전 문장의 주어를 지시어로 선택한다. (주어가 둘 이상일 경우 후행 주어를 선택)

예1) "시스템은 자판기이다."

"이것은 백 원이다."

∴ '이것'의 지시어는 '시스템'

예2) "자판기가 팔 수 있는 음료수는 커피, 콜라, 코코아이다."

"이것들은 모두 백 원이다."

∴ '이것들'의 지시어는 '커피, 콜라, 코코아' (복수/후행 나열)

현재 문장에서 지시대명사가 목적격일 경우 조건에 따라 지시어 결정. 지시어 후보가 둘 이상일 경우 후행 후보를 선택. 이때, 지시어 요건에 의해 지시대명사가 복수형일 경우에는 나열을 모두 지시어로 선택한다.

RULE(7.1) :

```
if !Fs &
   CrDP = Obj &
   PrP ∋ Obj
then PrP -> the last Obj
     CrDA -> Root(PrP)
```

이전 문장에 목적어가 있을 경우 : 이전 문장의 목적어를 지시어로 선택한다.

예) "이 시스템은 커피, 콜라, 코코아를 종이컵으로 판매하는 자판기이다."

"자판기는 이것들을 모두 백 원에 판매한다."

∴ '이것들'의 지시어는 '커피, 콜라, 코코아'

RULE(7.2) :

```
if !Fs &
```

CrDP = Obj &
PrP ≠ Obj &
PrP ∋ Subj
then PrP -> the last Subj
CrDA -> Root(PrP)

이전 문장에 목적어가 없을 경우 : 이전 문장의 주어를 지시어로 선택한다.

예) "이것은 스도쿠게임이다."
"이것을 실행하면 초기화면에서 게임설명과 게임시작, 종료 중 선택을 할 수 있다."
∴ '이것'의 지시어는 '스도쿠게임'

RULE(8.1) :
if !Fs &
CrDP = NPADV &
PrP ∋ Obj
then PrP -> the last Obj
CrDA -> Root(PrP)

현재 문장에서 지시대명사가 부사격일 경우 이전 문장의 목적어를 지시어로 선택. 이때 조응어 요건에 의해 지시대명사가 복수형일 경우에는 나열형을 모두 지시어로 선택한다.

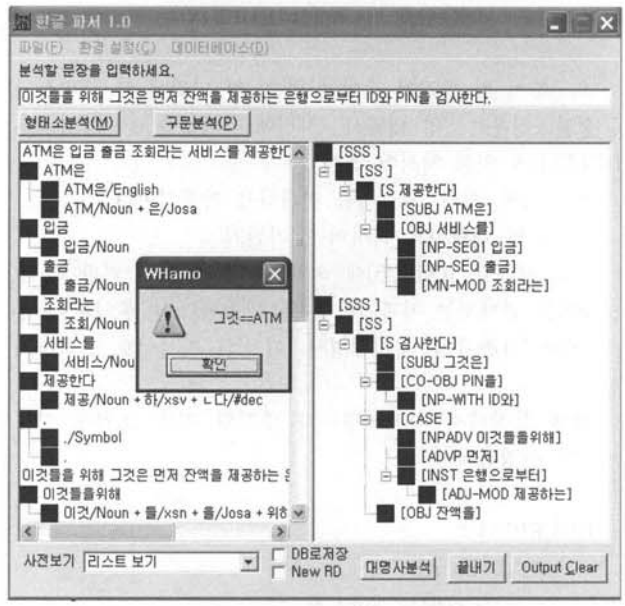
예) "ATM은 입금, 출금, 조회라는 세 가지 서비스를 제공한다."
"이것들을 위해 그것은 먼저 잔액을 제공하는 은행으로부터 ID와 PIN을 검사한다."
∴ '이것들'의 지시어는 입금, 출금, 조회
(* RULE(8)에 의해 처리됨.)
'그것'의 지시어는 'ATM'
(* RULE(6)에 의해 처리됨.)

위의 예에서 '그것'은 RULE(6.1)에 의해 'ATM'을 지시어로 선택하고, '이것들'의 지시어는 '세 가지 서비스'이지만 동격에 의해 수식되므로 지시어 요건에 따라 동격 '입금, 출금, 조회'로 확장된다.

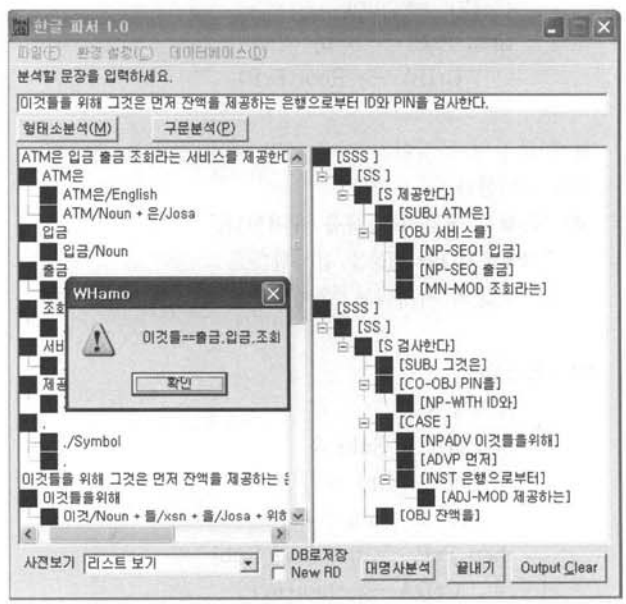
RULE(8.2) :
if !Fs &
CrDP = NPADV &
PrP ≠ Obj &
PrP ∋ Subj
then PrP -> the first Subj
CrDA -> Root(PrP)

이전 문장에 목적어가 없을 경우 선행하는 주어를 지시어로 선택한다.

(그림 5)와 (그림 6)는 RULE(8.1)의 예문에 대한 처리 결과를 보인다.



(그림 5) RULE(6)에 의한 조응어 처리 결과



(그림 6) RULE(8.1)에 의한 조응어 처리 결과

RULE(9.1) :
if CrDP = OF-MOD &
PrP ∋ Obj
then PrP -> the last Obj
CrDA -> Root(PrP)

현재 문장에서 지시대명사가 관형격일 경우 이전 문장에 목적어가 있으면 후행 목적어를 지시어로 선택한다.

예) "이 시스템은 커피, 콜라, 코코아를 종이컵으로 판매하는 자판기이다."
"이것들의 가격은 모두 백 원이다."

∴ '이것들'의 지시어는 '커피, 콜라, 코코아'

(그림 7)은 RULE(9.1)의 예문에 대한 처리 결과를 보인다.

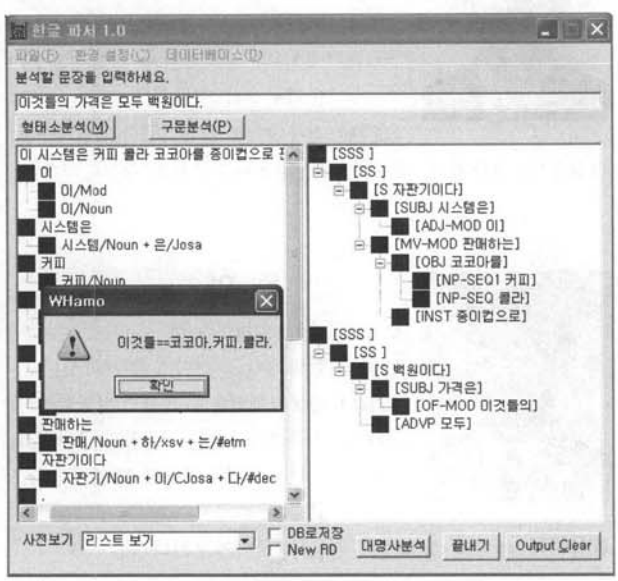
RULE(9.2) :

if CrDP = OF-MOD &
PrP ≠ Obj &
PrP ∋ Subj
then PrP -> the first Subj
CrDA -> Root(PrP)

현재 문장에서 지시대명사가 관형격일 경우 이전 문장에 목적어가 없으면 선행 주어를 지시어로 선택한다.

예1) "이것은 자판기이다."
"이것의 가격은 백 원이다."
∴ '이것'의 지시어는 '자판기'

예2) "자판기가 받을 수 있는 동전은 500, 100, 50, 10 원이다."
"이것의 반환레버를 작동해야지 거스름돈을 반환한다."
∴ '이것'의 지시어는 '자판기'



(그림 7) RULE(9.1)에 의한 조응어 처리 결과

5. 실험 및 평가

본 논문에서는 요구문서에서 지시대명사의 지시어 결정을 위한 경험 규칙을 기술하기 위하여 5개의 짧은 요구문서를 사용하였다. 요구문서는 연구[5, 12, 13]을 참고로 사용 가능한 다양한 패턴의 지시대명사를 추가하여 작성하였다. 각 요구문서의 문장 구성은 <표 1>과 같다.

<표 1>에 보인 요구문서를 통하여 경험 규칙 9개를 기술하였고, 이의 검증을 위하여 학부 소프트웨어공학 수업에서 조별 텀프로젝트로 학부생들이 기술한 요구문서 10개에 대해 실험을 하였다. 실험에 사용된 10개의 요구문서들의 문장 구성은 <표 2>와 같다.

<표 2>의 지시대명사 수는 영-대명사와 지시관형사를 포함하고 있다.

<표 3>은 본 연구에서 제안하는 시스템으로 <표 2>의 요구문서들의 지시대명사를 처리한 결과를 보인다.

본 연구의 실험 결과 전체 평균 재현율은 92.45%, 정확률은 69.68%의 성능을 보였다. <표 3>의 실험 결과에서 정확률이 현저히 떨어지는 경우의 원인은 크게 세 가지를 들 수 있다. 첫째, 문장이 복잡한 구조의 겹문장으로 구성된 경우로 홑문장 형태로 분리되면 정확률을 높일 수 있다. 둘째, 한 문장에 너무 많은 대명사를 사용한 경우로 사람이 직접 분석하기에도 다소 무리가 있는 경우였다. 셋째는 조응어의 범위를 벗어난 경우로 본 연구의 범위를 벗어난 경우였다.

또한, 정확률이 현저히 높게 나타는 경우는 홑문장 구조

〈표 1〉 요구문서의 구성

	어절수	문장수	문장별 어절수	지시대명사 수
RD1	143	17	8.4	10
RD2	61	8	7.6	4
RD3	54	7	7.7	7
RD4	179	23	7.8	8
RD5	256	18	14.2	9

〈표 2〉 실험에 사용된 요구문서의 구성

	어절수	문장수	문장별 어절수	지시대명사 수
RD1	218	27	8.1	19
RD2	467	34	13.7	26
RD3	237	25	9.5	9
RD4	270	16	16.9	14
RD5	135	14	9.6	7
RD6	132	16	8.3	28
RD7	147	12	12.3	12
RD8	192	15	12.8	19
RD9	99	14	7.1	11
RD10	122	10	12.2	12

〈표 3〉 재현율과 정확률

	재현율(%)	정확률(%)
RD1	94.74	44.44
RD2	100	38.46
RD3	66.67	16.67
RD4	100	92.86
RD5	85.71	100
RD6	85.71	70.83
RD7	100	100
RD8	100	78.95
RD9	100	63.64
RD10	91.67	90.91
평균	92.45	69.68

이면서 대명사가 매 문장마다 거의 동일한 문장성분으로 사용된 경우였다.

6. 결론 및 향후 연구

자연어 요구문서로부터 정형화된 요구사항을 자동으로 추출하기 위해서는 대명사의 특징을 고려하여 조응어를 처리하는 것이 매우 중요하다. 본 연구에서는 요구문서의 특성을 고려하여 대명사의 범위를 지시대명사로 제한하고, 지시어에 대한 요건을 정하여 조응어를 처리하는 경험 규칙을 기술하였다. 실험을 통해 지시어의 문장 범위를 5문장 정도로 높일 경우 정확률은 높아질 것이나 지시어 후보에 대한 조건을 보다 다양화해야 할 것으로 보인다. 또한, 관형어에 대한 조건을 추가하여 "이 중에서~"와 같이 관형어 뒤에 의존명사가 오는 경우를 개선해야 할 것이다.

향후 연구로서 정확률 개선을 위한 사항들을 보완하고, 요구사항을 자동으로 정확히 추출하기 위해서 관형어가 지시하는 보통명사(예, 시스템, 프로그램 등)는 요구문서에서 특별한 의미를 부여받지 못하므로 보어나 의미적으로 부합되는 조응어로 대체가 필요하며, 대량의 전문 요구문서를 통해 성능을 개선해야 할 것이다.

참 고 문 헌

[1] 김상수, 김계성, 노태길, 이상조, "문서요약을 위한 조응 대용해결," 2002년도 한국정보과학회 가을 학술발표 논문집, Vol.29. No.02, pp.679-681, 2002.

[2] 조은경, 서정연, "대화 시스템에서의 조응어 해석," 제16회 한글 및 한국어 정보처리 학술대회 논문집, 제16권 제1호, pp.283-289, 2004.10.

[3] Ki-Seon Park, Keunyong Lee, Moon-Kun Lee, Dong-Un An, Yong-Seok Lee, "Antecedent Decision Rules for Anaphora Resolution of Natural Language Requirements Document in Korean," 12th International Conference on Human-Computer Interaction (HCI International 2007), pp.598-602, 2007.07.

[4] 박갑수, 조규빈, "고교 문법 자습서," 지학사, 1993.

[5] 강인혜, 양진석, "초보자를 위한 Esterel 프로그래밍," 홍릉과학출판사, 2005.

[6] 강승식, 윤보현, 우종우, "Coreference Resolution을 위한 3인칭 대명사의 선행사 결정 규칙," 한국정보처리학회 논문지 B, Vol.11-B. No.02, pp.227-232, 2004.04.

[7] Antonio Ferrández, Jesús Peral, "A computational approach to zero-pronouns in Spanish," In Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics (ACL2000), Hong-Kong, China, pp.166-172, October 2000.

[8] 노지은, 나승훈, 이종혁, "중심화 이론을 이용한 텍스트 구조화," 한국정보과학회 논문지 B, Vol.34. No.06, pp.572-583, 2007. 06.

[9] 노지은, 이종혁, "구문 정보와 비용기반 중심화 이론에 기반한 자연스러운 지시어 생성," 정보과학회논문지 B, Vol.31. No.12, pp.1649-1659, 2004.12.

[10] 차건회, 송도규, 박재득, "한국어 대용과 생략 해결을 위한 센터링 이론의 적용," 제9회 한글 및 한국어 정보처리 학술대회, 한국정보과학회 언어공학연구회, pp.347-352, 1997.10.

[11] 홍민표, "센터링 이론과 대화체에서의 논항 생략 현상," 인지과학 제11권 제1호, pp.9-24, 2000.03

[12] 윤청, "성공적인 소프트웨어 개발 방법론," 생능출판사, 1999.

[13] Beum-Seuk Lee, Barrett R. Bryant, "Automated Conversion from Requirements Documentation to an Object-Oriented Formal Specification Language," Proceedings of the 2002 ACM symposium on Applied computing, Madrid, Spain, pp.932-936, 2002.03.

[14] 이현영, 이용석, "내포문의 단문 분할을 이용한 한국어 구문 분석," 한국정보과학회 논문지 B, Vol.35. No.01, pp.50-58, 2008.01.

박 기 선

e-mail : icarus@jbnu.ac.kr

2003년 전북대학교 컴퓨터정보학과(이학석사)

2003년~2004년 전북대학교 영상정보신기술연구센터 연구원

2007년 전북대학교 컴퓨터공학과(박사수료)

관심분야 : 정보검색, 자연어처리, 한국어정보처리 등

안 동 언

e-mail : duan@jbnu.ac.kr

1981년 한양대학교 전자공학과(학사)

1987년 한국과학기술원 전산학과(공학석사)

1995년 한국과학기술원 전산학과(공학박사)

1995년~현 재 전북대학교 IT정보공학부 교수

관심분야 : 정보검색, 자연어처리, 한국어정보처리 등

이 용 석

e-mail : yslee@jbnu.ac.kr

1977년 서울대학교 전자공학과(학사)

1979년 한국과학기술원 전산학과(이학석사)

1995년 일본 국립도쿠시마대학교 지능정보학과(공학박사)

1979년 한국표준연구소 선임연구원

1983~현 재 전북대학교 컴퓨터공학부 교수

관심분야 : 인공지능, 자연어처리, 영한기계번역 등