

한국어 대명사 및 한정 명사구에 대한 상호참조해결

박천음^o, 최경호, 이흥규, 이창기

강원대학교, 컴퓨터학과

{parkce, gangsparkle, proname, leeck}@kangwon.ac.kr

Coreference Resolution for Korean Pronouns and Definite Noun Phrases

Cheon Eum Park^o, Kyoung Ho Choi, Hong Gyu Lee, Chang Ki Lee
Kangwon National University

요 약

본 논문은 Stanford의 다 단계 시브(Multi-pass Sieve) 상호참조해결을 기반으로, 한국어에 적용한 한국어 상호참조해결(선행 연구)을 이용하여 한정 명사구에 대한 처리와 확장된 대명사 상호참조해결 방법을 제안한다. 지시 관형사와 명사가 결합하여 형성되는 한정 명사구는 일반 멘션(mention)의 특징과 대명사 속성을 한 번에 갖게 된다. 이렇게 되면, 한정 명사구는 모든 시브(sieve)에서 상호참조를 진행할 수 있게 된다. 따라서 이런 특징으로 한정 명사구를 어떤 관점(멘션 또는 대명사)에서 상호참조해결하는 것이 좋은지 보인다. 또한 이런 한정 명사구의 대명사 속성을 이용하기 위해 문법적·의미적 규칙을 적용할 것을 제안한다. 그 결과, 본 논문의 선행 연구인 한국어 상호참조해결에 비하여 CoNLL 값이 약 0.8%만큼 향상되어 61.45%를 측정하였다.

주제어: 한국어 상호참조해결, 대명사 상호참조해결, 한정 명사구 상호참조해결

1. 서론

질의 응답(Question and Answering)에서 보통 질문의 물음은 “이것은 무엇인가?”, “이 사람의 이름은?”, “이 동물이 상징하는 것은 무엇인가?” 등과 같이 대명사(pronouns)나 한정 명사구(definite NPs라 하며, 즉 지시관형사가 붙은 명사구)가 포함된 문장이 된다. 이와 같은 질문에 대한 문서 검색에서, 이런 대명사나 한정 명사구가 가리키는 것이 무엇인지 파악되어야 정확한 검색을 할 수 있다. 즉, 마지막 물음에 나타나는 대명사 또는 한정 명사구가 전체 질문의 핵심 키워드가 되기 때문에, 올바른 답변을 하는데 많은 도움이 된다. 이처럼 대명사 및 한정 명사구가 어떤 선행사를 가리키는지에 대한 분석은 상호참조해결(coreference resolution)로 해결한다. 만약 이런 대명사분석이 해소되지 않는다면, 문장 전체를 분석하여 핵심 키워드(즉, 중심어)를 추출하고, 이를 정리하여 질문의 요지를 추론해야 한다. 그렇기 때문에, 이와 같은 대명사 및 한정 명사구의 해결은 반드시 필요한 부분이다. 게다가 이러한 대명사 해결의 중요성 때문에 CoNLL-2011에서 이미 영어에 대한 상호참조해결을 주제로 다뤘었다.

한국어에서 대명사는 앞서 등장한 명사에 대하여 다시 언급할 때 그 대상을 대신할 단어를 말한다. 그리고 지시관형사(demonstrative pronoun)는 어떤 대상을 지시하여 가리키는 관형사를 말하며, 이 관형사는 선행하는 대상의 단어보다 좀더 상위 개념의 단어(예를 들어, “기가 큰 당나귀 ...”, “이 동물은 ...”과 같이 여기서 두 번째 문장에 나타난 범주어 [동물]은 첫 번째 문장의 [당나귀]보다 상위어다) 또는 같은 단어와 함께 사용된다. 이러한 언어적 특성에 대하여 사람은 직관에 의해 쉽게 이해할 수 있지만, 기계는 그렇지 않다. 따라서 기계가 이러한 언어적 특성을 이해하기 위하여 두 가지 패러다임(paradigm)이 적용 되는데, 먼저 대용어해결(anaphora resolution)과 다른 하나는 상호참조해결이다.

본 논문에서는 한국어 대명사와 한정 명사구에 대하여 좀더 실질적으로 해결할 수 있는 상호참조해결을 적용하였다. 이와 같이 상호참조해결을 이용한 대명사 및 한정 명사구 해결은 정보 검색, 정보 추출, 문서 요약, 이벤트 추출 및 탐지, 영형(zero pronouns) 복원 그리고 질의응답 등에 응용될 수 있다.

2. 관련 연구

기존의 대명사에 대한 해결 방법으로는 주로 대용어처리가 사용되어 왔다. 이중 크게 세 가지 모델을 이용해 왔는데, 바로 중심화 이론(centering theory)[4]과 Hobbs 알고리즘(Hobbs algorithm)[5], RAP(Resolution of Anaphora Procedure) 알고리즘[6]이다.

특히 두 모델(중심화 이론, Hobbs 알고리즘)은 각각에 대하여 문장이 길어지거나 문법구조가 복잡해지는 경우, 대명사 해결이 어려워진다. 그리고 영어권에서 제안된 것이라 한국어에 맞춰 재해석하기에 어려움이 있으며, Hobbs는 문법 트리에 기반하고 있다. 그러나 RAP 알고리즘은 문법의 복잡함에 있어 독립적이며, 본 논문에서 사용하는 의존 구문 문법에 적용하기에 어렵지 않다. 더하여 RAP 알고리즘의 특성은 중심화 이론에서 중요시 여기는 몇 가지 요소(즉, grammatical role, recency, repeated mention, current discourse topic, parallelism)[3]들을 내포하며, 문법 체계에서의 특징을 두드러지게 표현하여 사용하였다(즉, 문법 구조 상 각 특징들을 점수화 하였다). 이에 따른 추가적인 설명은 4장을 참조한다.

앞서 언급한 세 가지 모델들의 공통점은 바로 주어의 중심어(head)가 문장에서 가장 큰 영향력이 있다고 보고 대용어 해결을 진행하는 것이다. 그렇다면 상호참조해결도 이와 유사한 개념으로 진행할 수 있다. 이를 뒷받침하여, Stanford의 [1]에서는 위와 같은 개념과 entity centric 모델, 대명사 해결에 대하여 대명사의 속성과 개체명을 이용한 해결 방법을 영어에 적용하여 상호참조를 해결하였고 Closed Track으로 CoNLL-F1 값 57.8%의 성능을 보였다. 이에 따라 본 논문의 선행 연구인 [2]에서, Stanford의 [1] 중 한국어에 적합한 방법들만 추려내어 적용하여 CoNLL-F1 60.65%를 측정하였으며, 이와 같은 방법이 한국어에도 적합하다는 것을 보였다. 그러나 [2]에서는 지식기반 사전을 참조하지 않아 상호참조해결에 대한 약간의 어려움이 따랐다. 따라서 본 논문에서는 이런 중심어에 대한 개념과 [2]의 방법을 개선하는 것과 세종대그셋에서 추출한 지식기반 데이터를 이용하여, 문장 간(inter-sentential) 또는 문장 내에서 등장하는 선행사 후보들과 이에 참조되는 대명사에 대한 해결을 제안한다.

그 외로 CogNIAC의 경험규칙을 이용하여 한국어 3인칭 대명사를 해결한 [8]이 있다. [8]은 전체 평균 재현율(recall) 79.0%, 정확률(precision) 86.8%를 보인다. 그러나 본 논문과 달리 [8]은 한국어 3인칭 대명사에 대한 상호참조해결로 한정된 해결 방법이다. 이와 달리, 본 논문은 세종대그셋에서 추출한 한국어의 모든 대명사와 한정 명사구에 대한 해결을 제안한다.

본 논문의 구성은 다음과 같다. 3장에서 기존에 다뤘던 한국어 상호참조해결에 대하여 설명하고 4장에서 RAP 알고리즘의 처리 방법 및 과정을 보인다. 이어서 5장에서는 본 논문에서 제안하는 대명사 및 한정 명사구의 상호참조해결 방법에 대해서 다루며, 6장에서 각 실험에 대한 방법과 결과를 보인다. 그리고 7장을 마지막으로 결론을 맺는다.

3. 한국어 상호참조해결

자연스러운 문맥을 위해 사람들은 먼저 말했던 단어를 다시 말할 때, 대명사를 이용하여 같은 말의 반복을 피한다.

보통 대명사를 해결하기 위해서는 두 가지 패러다임이 적용된다. 대용어해결과 상호참조해결이 그것인데, 이 둘은 개념적으로 거의 유사하여 모호하지만, 그 방법에 차이가 있다. 즉, 목적은 둘 다 대명사에 대응되는 선행사들을 찾는 것이다. 하지만 대용어해결은 선행사와 대용어와의 관계에 있어 전이적인 특성만 갖는다. 반면에 상호참조해결은 전이적, 대칭적, 재귀적 등과 같이 3가지의 특성을 갖고 참조해결을 진행한다(즉, $a \rightarrow b$, $b \rightarrow c$ 이면 $b \leftarrow c$, $a \leftarrow b$ 이며, $c \leftrightarrow a$ 와 같다)[3]. 즉, 대용어해결은 단어 간의 첨자(index)가 각각에 대하여 연쇄적으로 해결되고 상호참조해결은 첨자가 개체 별로 중복 집합관계에 따른다.

대용어 해결과 상호참조 해결은 이와 같은 차이를 보이며, 본 논문에서는 상호참조 해결을 기반으로 대명사 및 한정 명사구 해결을 진행한다. 한국어에 대한 상호참조 해결[2]은 본 논문의 선행연구로서, Stanford의 [1]을 기반으로 한국어에 적용 가능한 방법들과 본 논문에서 다루는 뉴스 도메인과 질의 응답 도메인에 적합한 방법들만 선택하여 적용하였다. 이 방법은 크게 3단계로 멘션 탐지(Mention Detection)와 다 단계 시브, 후처리(Post Processing)로 구성되며 해결 과정은 [표 1]과 같다.

표 1. 상호참조해결 순서

Pass	Method	
Pass 1	Mention Detection	
	Sieve 1	Exact String Match
	Sieve 2	Precise Constructs
Pass 2	Sieve 3-5	Strict Head Match A-C
	Sieve 6	Proper Head Noun Match
	Sieve 7	Relaxed Head Match
	Sieve 8	Pronoun Resolution
Pass 3	Post Processing	

본 논문은 선행연구인 [2]를 기반으로 대명사와 한정 명사구에 대해 좀 더 초점을 맞추고 한국어에 대한 상호참조해결 성능을 더욱 개선시켰다.

4. RAP 알고리즘

RAP 알고리즘은 문장 내 또는 문장 간의 3인칭 대명사와 어휘 대용어(lexical anaphors)에 대한 선행사를 찾아 해결한다[6]. 이 알고리즘은 앞서 언급한 요소들에 대한 개념을 포함시켜 정의하고 조건들을 제안하였다. 또한 이런 요소적 특징들을 점수화 하여 가장 높은 점수를 가진 선행사(또는 멘션)와 대명사

해결을 진행하며, 문법적 특징에 따른 점수 가중치는 [표 2]와 같다.

표 2. 요소의 특징에 따른 가중치

Factor type	weight
(1) Sentence recency	100
(2) Subject emphasis	80
(3) Existential emphasis	70
(4) Accusative emphasis	50
(5) Indirect object and oblique emphasis	40
(6) Head noun emphasis	80
(7) Non-adverbial emphasis	50

RAP 알고리즘은 입력된 모든 문장에 대하여 가중치를 부여한다. 알고리즘의 첫 번째 단계는 현재 등장한 문장에 존재하는 요소들에 대하여 가중치를 반으로 나눈다. 이것은 등장하는 대명사로부터 앞서 등장한 문장들과의 거리 차이를 정의할 수 있다. 두 번째로 현재 문장에 등장하는 멘션들에 대하여 [표 2]를 이용하여 문법적 요소가 맞는 기준의 가중치를 부여한다. [표 2]의 각 요소들에 대하여 살펴보면 다음과 같으며[3], 이런 방법을 본 논문의 대명사 상호참조해결에 적용해 보고 본 논문의 제안 방법과 비교해보았다.

(1)은 현재 등장한 문장에 대한 가중치로 현재 문장 내의 모든 멘션들에 가중치 100을 더한다. (2)는 현재 진행 중인 문장에서 주어인 멘션에 가중치를 더한다. (3)은 영어의 “There be...”와 같은 유도부사에 대한 특징이다. 영어에서는 해당 부분이 멘션으로 등장하면 가중치를 더하지만, 한국어에서는 이와 같은 문법 구조가 없기 때문에 사용하지 않는다. (4)는 직접 목적어가 등장하면 가중치를 더하며, (5)는 간접 목적어가 등장할 경우 가중치를 더한다. 그러나 역시 한국어에는 직접 목적어와 간접 목적어의 구분이 명확하지 않다. 따라서 (5)에 대한 가중치는 생략한다. (6)은 해당 멘션의 중심어가 존재할 경우, 즉 하나의 명사구에서 중심어가 되는 명사를 수식하는 명사들이 아니라, 그 명사구의 중심어에 대한 멘션인 경우 가중치를 부여한다(예를 들어, “르노 삼성 자동차”인 명사구에서 중심어인 “자동차”를 포함한 멘션 [르노 삼성 자동차]에만 가중치를 부여하고 이 중심어를 수식하는 [르노]와 [삼성] 멘션에는 부여하지 않는다). (7)은 부사격에 대한 가중치 부여이다.

5. 대명사 상호참조해결

본 논문의 대명사 상호참조해결은 Stanford의 [1]을 기반으로 한국어에 적용한 [2]로부터 확장시켰다.

5.1 한국어 대명사 및 한정 명사구

대명사는 앞서 언급된 명사에 대해 다시 언급할 때 사용된다. 한국어에서의 대명사는 영어와 달리 보다 더 세분화 되고 의미가 다양하다(예를 들어, 인칭/비인칭 대명사 모두 지시하는 대상 간의 거리에 따른 속성과 존칭에 관한 속성 등이 있다). 본 논문에서는 세종 코퍼스에서 추출한 총 188개의 대명사를 대명사 사전으로 사용하였다.

한정 명사구는 지시 관형사와 명사(한정 명사구에서 중심어)가 함께 사용되는 문법 구조이다. 특히 한정 명사구의 중심어는 지시 관형사가 선행사를 가리키고 있으므로, 지시하는 대상보다 상위어가 사용되는 특징을 보인다. 이에 따라, 한정 명사구를 해결할 때, 대명사 해결보다 자질을 더 다양하게 사용할 수 있다. 즉, 지시 관형사 중 대부분은 대명사와 같은 단어들이 사용된다.

따라서 본 논문에서는 이런 단어들에 대해 대명사 속성을 부여하고, 한정 명사구에 포함된 중심어를 이용하여 개체명 정보를 확인할 것을 제안한다. 그 외에도 세종 코퍼스로부터 시소

러스 정보를 추출하여 의미 사전(즉, semClass)을 만들고, 이것을 한정 명사구의 중심어에 적용하여 중심어 의미 매치(head semantic match)를 수행하여 대명사 및 한정 명사구의 해결을 제안한다.

5.2 대명사 상호참조해결 방법 확장

본 논문에서 제안하는 대명사 상호참조해결 알고리즘은, 문장 규칙과 중심화 이론의 전이적 특성을 적용시켜, 그 특징들에 따라 가중치를 부여하는 방법인 [2]을 대명사 속성 및 한정 명사구의 특징에 따른 방법으로 확장시킨 것이다. 이에 대한 방법은 다음과 같다.

- 대명사 문자열 매치(pronoun string match)
- 선행사와 한정 명사구의 중심어간의 문자열 매치(head string match)
- 선행사와 한정 명사구의 중심어간의 의미 매치(head semantic match)
- 재귀 대명사는 현재 문장의 주어와 상호참조한다.

위의 규칙들 중에서 첫 번째 규칙은 선행 멘션 중 현재 대명사와 문자열이 같은 대명사 멘션에 가중치를 부여한다. 그 이유는 보통 한 번 대명사로 언급한 것은 또다시 같은 대명사로 언급하기 때문이다. 다음으로 두 번째 규칙은 선행사가 수식어들로 인하여 의미가 명확해진 범주어에 대한 해결 방법이다(예를 들어, “한국 축구 국가대표의 상징인 호랑이는 ...”. “그 호랑이는 ...”일 경우 [그 호랑이]에서의 중심어와 선행사와의 중심어는 서로 같다). 그리고 세 번째 규칙은 선행사의 중심어와 한정 명사구의 중심어 사이에서 서로의 개체명을 이용하여 유사한 의미를 나타내는지 확인하고, 의미사전을 이용하여 서로 계층관계인지 파악한다. 마지막으로 재귀 대명사는 해당 문장의 주어를 참조한다. 그리고 여기에 더하여, 대명사가 선행사를 참조하지 않는 경우(즉, 후방 조응사)에 대비하여 대명사 상호참조해결 알고리즘에 임계치를 적용하였다. 따라서 이 임계치를 넘는 대명사만이 상호참조가 가능하다.

6. 실험 및 결과

본 논문은 [2]에서 사용하였던 평가 데이터(news 20문서 + wiseqa 205문서, 총 225문서)를 그대로 사용하였으며, 평가 데이터는 ETRI의 언어분석기를 이용하였다. 2명의 연구원이 추가로 상호참조해결 데이터를 구축하였으며, 총 3명의 연구원간의 데이터 일치도를 구하였다. 일치도를 구하는데 사용된 지표는 Cohen의 카파 계수(Cohen’s kappa coefficient)[9]를 사용하였다.

카파 계수는 2명의 평가자 간의 의견이 얼마나 일치하는지에 대한 평가이다. 따라서 일치도는 2명의 데이터 결과를 서로 겹쳐가며 총 3번(즉, 연구원: 1, 2, 3 중, 실험 A: 1-2, B: 2-3, C: 3-1) 측정하였다. 이에 따른 계수 값은, 두 명의 연구원간에 서로 동일한 문서 중에서 하나의 엔티티를 기준으로 하여 비교를 통해 구한다. 여기서 멘션의 비교 기준은 중심어로 하여 문장의 인덱스와 멘션의 중심어 인덱스가 서로 같으면 의견이 일치한 것으로 본다. 이와 같이, 계산한 문서 당 계수를 더하고 평균을 내어 카파 계수를 측정한다.

표 3. 상호참조해결 카파 계수 일치도

Kappa coefficient	54.16%	55.65%	73.63%	61.12%
-------------------	--------	--------	--------	--------

[표 3]에 따라 본 논문의 카파 계수는 61.12%로 측정되었으며, 카파 계수 지표 ‘0~20%(근소한 일치)’, ‘21~40%(적당한 일치)’, ‘41~60%(평균 일치)’, ‘61~80%(상당한 일치)’, ‘81~100%(완벽히 일치)’ 중 61~80%에 속해 상당한 수준의 일치도를 보였다.

6.1 평가 방법

본 논문에서 적용한 평가 방법은 [2]에서 사용하였던 MUC과 B^3 , CEAF-e, CoNLL 등을 그대로 사용하였으며, CoNLL은 앞의 세 지표의 F1 값에 대하여 평균(mean)을 낸 것이다.

6.2 실험 결과

본 논문은 선행 연구인 [2]을 기반으로 대명사 해결을 한국어에 적합하도록 확장시킨 것이다. 이에 따라, 본 논문에서 제안하는 것이 적합한 방법인지 다음 실험을 통하여 성능을 보인다. 우선 [표 4]는 한정 명사구를 어떤 관점(즉, 멘션 또는 대명사)에서 해결하는 것이 좋은지 확인한다. 즉, 한정 명사구에 대하여, 이것을 멘션으로 보아 모든 시브를 이용하여 상호참조를 해결하는 것인가, 아니면 이것을 대명사로 보아 대명사 상호참조해결 시브에서만 처리하는가에 대하여 보인다. [표 4]의 재현율(recall)과 정확률(precision) 또한 F1과 마찬가지로 성능지표 세 가지(즉, MUC, B^3 , Ceaf-e)를 평균한 것이다.

표 4. 한정 명사구의 처리 방법에 따른 성능

Previous paper	64.2%	57.57%	60.65%
Multi-pass Sieve	60.74%	61.22%	60.96%
Pronouns match Sieve	59.39%	61.32%	60.33%

[표 4]는 [2]를 기반으로 하여 한정 명사구의 추출과 해결을 추가하여 실험한 것이다. 여기서 Multi-pass Sieve는 한정 명사구에 대하여 일반 멘션과 대명사 속성을 포함하고, 모든 시브를 이용하여 처리한 경우이다. 반면에, 대명사 시브(Pronouns match Sieve)는 한정 명사구를 오직 대명사로 보고 대명사 상호참조해결에서만 처리한 경우이다. 결과적으로, 한정 명사구는 멘션과 대명사 속성을 갖게 하고, 모든 시브를 이용하여 해결하는 방법이 더 우수한 성능을 기록하는 것을 알 수 있다. 따라서 앞으로의 실험은 모두 한정 명사구를 모든 시브에 대하여 처리하였을 경우에 기반 하여 진행한다.

그 다음으로, [그림 1]은 최적의 가중치 파라미터(50 / 20 / 40 / 20: 61.45%)를 이용하여, 한정 명사구를 해결한 것과 그렇지 않은 것에 대한 차이를 비교하고 한정 명사구에 대한 사용을 증명한다.

[그림 1]의 그래프는 먼저 선행 연구의 성능을 보인다. 선행 연구에서는 한정 명사구에 대한 처리가 없었다. 즉, 기본적으로 한정 명사구가 일반 멘션으로 탐지되고 그 안의 지시관형사가 대명사로 추출되었다. 이런 경우, 한정 명사구와 지시관형사는 어떤 규칙에 제한되지 않아 특정 기준에 의한 참조에서 벗어날 수 있다. 따라서 두 번째의 성능 그래프는 이런 경우(즉, 한정 명사구와 지시관형사의 참조)를 제거한 것에 대한 상호참조해결 성능을 보인다. 세 번째 성능 그래프는, 본 논문에서 제안한 방법 중 먼저 대명사 해결 방법만 적용하여 나타난 성능이다. 그 다음으로, 한정 명사구만 해결하는 경우에 대한 성능을 보인다. 마지막으로 보인 성능은 한정 명사구와 대명사 해결 방법 모두를 수행했을 경우에 나타난 것이다.

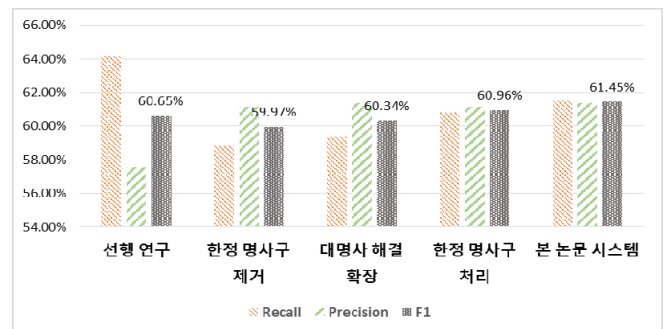


그림 1. 각 대명사 해결 방법에 따른 성능 비교

[그림 1]에서 첫 번째 그래프(선행 연구)는 정확률보다 재현율이 더 높게 측정되었다. 그에 반해, 한정 명사구를 모두 제거한 두 번째 그래프에서는 정확률이 더 높아졌다. 이와 같은 이유는, 한정 명사구와 한정 명사구 안에 있는 지시 관형사(대명사로 멘션이 잡혔었다)가 모두 빠지면서, 정말 대명사로 쓰이는 멘션만 남았기 때문이다. 그리고 세 번째 그래프는, 본 논문에서 제안한 대명사 해결 방법만 적용한 것으로, 재현율이 다소 향상된 것을 확인할 수 있다. 그 다음은 한정 명사구에 대해서만 성능을 측정하는 것으로, 한정 명사구를 멘션으로 추출하고 [표 4]와 같이 모든 시브에 적용하여 상호참조해결을 진행했을 경우, 재현율과 정확률이 서로 균형을 맞추는 것을 볼 수 있다. 마지막으로, 본 논문에서 제안한 방법들을 모두 적용하였을 경우에 CoNLL 61.45%로 선행 연구에 비해 약 0.8% 향상된 것을 확인할 수 있다.

최종적으로, 4장에서 언급한 RAP 알고리즘과 본 논문에서 사용하는 대명사 가중치 알고리즘에 대한 성능 비교를 [표 5]에서 보인다. [표 5]은 한정 명사구를 추출하고 모든 시브에서 상호참조가 발생한 것을 기반으로 한다. 즉, 대명사 상호참조 해결 시브에서 사용된 알고리즘만 변경하여 성능을 측정하는 것이다. [표 5]의 성능 기준은 [표 4]와 같다.

표 5. 알고리즘의 성능 비교

Model	Recall	Precision	CoNLL-F1
RAP algorithm	61.76%	60.23%	60.98%
This paper	61.54%	61.38%	61.45%

[표 5]의 결과로, 본 논문에서 제안한 대명사 상호참조해결 방법이 약 0.47% 더 높게 측정되었다. 이와 같은 이유는, RAP 알고리즘의 가중치 적용 방법은 문법 규칙에만 의존적이다. 또한 영어에 최적화되었기 때문에 한국어에 적용할 수 없는 규칙(유도부사, 간접 목적어 등)도 있다. 이에 반해 [2]는 개체명 정보를 이용한 대명사 상호참조해결을 수행한 [1]을 기반으로, 좀 더 다양한 개체명 정보들을 적용하고 분류하여 한국어에 더욱 정확한 상호참조를 가능하게 했고, 본 논문에서는 이것의 사용을 더욱 확장하여 대명사뿐만 아니라 한정 명사구에도 적용하였다. 또한 한정 명사구의 명사구 특징(즉, 일반 멘션으로 사용)과 대명사의 속성정보(즉, 대명사 시브에서 참조해결)를 이용하였다. 따라서 본 논문에서 제안한 방법이 기존 RAP 알고리즘 보다 한국어에서 좋은 성능을 보인 것을 확인할 수 있었다.

7. 결론

본 논문은 선행 연구인 [2]를 기반으로 하여 더 나은 상호참조 해결 성능을 위해 한국어 대명사 및 한정 명사구 상호참조해결 알고리즘을 제안하였다. 이에 따라, 한정 명사구 처리 방법과 대명사 해결에 대한 가중치 파라미터의 최적 조합을 보였고, 이것을 기반으로 본 논문에서 제안한 알고리즘의 성능 비교 실험을 진행 하였다. 그 결과, 상호참조해결의 최종 성능(대명사 상호참조해결)에 대한 CoNLL 값이 61.45%로 선행 연구인 [2]에 비해 약 0.8% 향상됨을 보였다. 또한 RAP 알고리즘과 본 논문에서 제안한 대명사 상호참조해결 알고리즘을 한국어에 대해 비교했을 경우, 각각 60.98%와 61.45%로 본 논문에서 제시한 방법이 0.47% 더 높았음을 보였다. 이에 따라 본 논문에서 제안한 방법이 한국어 상호참조해결의 성능 향상에 도움이 된다는 것을 알 수 있다.

그러나 여전히 상호참조해결에 대한 어려움이 남아있다. 상호참조해결은 사람 조차 직관으로 해결하기 힘든 모호성이 있는데, 예를 들어 “한국 축구 국가대표의 상징인 동물은 호랑이다.”와 같은 문장에서 [한국 축구 국가대표의 상징인 동물]

은 [호랑이]로 전이될 수 있다. 그러나 [호랑이]는 그 의미가 선행한 멘션만 나타내는 것이 아니라, 동물의 호랑이가 될 수도 있기 때문에 상호참조해결의 재귀적 특성을 만족할 수 없다. 이것뿐만 아니라 [통일신라]와 [그 당시]라는 멘션들이 등장한다면, 여기서 [통일신라]는 그때의 시대를 나타내는 단어인지, 아니면 나라를 나타내는 단어인지에 대한 의미가 모호하기 때문에 [그 당시] 멘션과 상호참조가 가능한지 불분명하다.

또한 “그 후”, “그 외”, “그 결과”와 같은 한정 명사구에서 지시 관형사는 선행하는 멘션을 가리키지만 한정 명사구 자체의 의미는 그 다음에 나타날 명사구(멘션)를 가리킨다. 즉, 후방 조용사(cataphora)로 본 논문에서는 적용하지 않았다. 그렇기 때문에 위와 같은 경우의 대명사에 대한 상호참조해결은 불가능하였다.

이런 문제점들을 해결하기 위해서 향후 계획으로, 개체 중심 상호참조해결에서 엔티티에 대한 재정의가 필요하다. 즉, world knowledge와 문서내의 정보를 적절하게 이용하여 상호참조하는 방법에 대해 재검토하여 일관성 있고 올바른 기준을 정의할 것이다. 또한 그 정의에 따라 상호참조해결 데이터를 추가적으로 더 구축하여 다양한 실험을 진행할 것이다. 여기에 더하여 관계 추출 등을 이용하여 개체 간의 의미를 명확하게 할 예정이다. 그리고 앞서 언급한 후방 조용사에 대한 대명사 해결을 진행할 것이다.

감사의 글

본 연구는 미래창조과학부 및 한국산업기술평가위원회의 산업융합원천기술개발사업(정보통신)의 일환으로 수행하였음 [10044577, 휴먼 지식증강서비스를 위한 지능진화형 WiseQA 플랫폼 기술 개발]

참고문헌

- [1] Heeyoung Lee, Angel Chang, Yves Peirsman, Nathanael Chambers, Mihai Surdeanu & Dan Jurafsky. “Deterministic coreference resolution based on entity-centric, precision-ranked rules,” Association for Computational Linguistics, 2013.
- [2] 박천음, 최경호, 이창기, “Multi-pass Sieve를 이용한 한국어 상호참조해결”, 정보과학회논문지(Journal of KIISE, JOK) 등재 예정, 2014.
- [3] Manfred Stede. “Discourse processing,” Synthesis Lectures on Human Language Technologies, 4.3: 1-165, 2011.
- [4] Barbara J. Grosz, Scott Weinstein & Aravind K. Joshi. “Centering: A framework for modeling the local coherence of discourse.” Computational linguistics 21.2, pp. 203-225, 1995.
- [5] Hobbs Jerry R. “Resolving pronoun references.” Lingua 44.4 pp. 311-338, 1978.
- [6] Lappin Shalom and Herbert J. Leass. “An algorithm for pronominal anaphora resolution.” Computational linguistics 20.4 pp. 535-561, 1994.
- [7] Brennan Susan E., Marilyn W. Friedman and Carl J. Pollard. “A centering approach to pronouns.” Proceedings of the 25th annual meeting on Association for Computational Linguistics. Association for Computational Linguistics, 1987.
- [8] 강승식, 윤보현, 우종우, “Coreference Resolution을 위한 3인칭 대명사의 선행사 결정 규칙”, 정보처리학회논문지B 11권 2호, pp. 227-232, 2004.
- [9] Cohen, Jacob. “Weighted kappa: Nominal scale agreement provision for scaled disagreement or partial credit.” Psychological bulletin 70.4 pp. 213, 1968.