KIPS Tr. Software and Data Eng.
Vol.3, No.10 pp.407~420 pISSN: 2287-5905

http://dx.doi.org/10.3745/KTSDE.2014.3.10.407

Korean Homograph Tagging Model based on Sub-Word Conditional Probability

Shin Joon Choul† · Ock Cheol Young††

ABSTRACT

In general, the Korean morpheme analysis procedure is divided into two steps. In the first step as an ambiguity generation step, an Eojeol is analyzed into many morpheme sequences as candidates. In the second step, one appropriate candidate is chosen by using contextual information. Hidden Markov Model(HMM) is typically applied in the second step. This paper proposes Sub-word Conditional Probability(SCP) model as an alternate algorithm. SCP uses sub-word information of adjacent eojeol first. If it failed, then SCP use morpheme information restrictively. In the accuracy and speed comparative test, HMM's accuracy is 96.49% and SCP's accuracy is just 0.07% lower. But SCP reduced processing time 53%.

Keywords : Korean Morphological Analyzer, HMM, Homograph, Tagging

부분어절 조건부확률 기반 동형이의어 태깅 모델

신 준 철† · 옥 철 영††

요 약

한국어 형태소 분석 및 태깅은 크게 2가지 단계로 나뉜다. 첫 번째 단계는 어절을 분석하여 후보들을 생성하는 것으로, 여러 의미를 가진 어절은 이 단계에서 다양한 후보들이 생성된다. 두 번째는 문맥 정보를 이용하여 후보 중에 가장 적절한 하나를 선택하는 단계로, 흔히 태깅이라 한다. 일반적으로 두 번째 단계에서는 은닉 마르코프 모델(Hidden Markov Model, 이하 HMM)을 자주 사용하지만, 본 논문에서는 처리속도를 향상시킨 부분어절 조건부확률 모델을 제안한다. 이 모델은 우선적으로 인접 어절 정보를 이용하여 현재 처리 중인 어절의 의미를 결정하고, 예외적으로 용언이 인접한 경우에만 후보 정보의 극히 일부분을 이용한다. 실험 결과 정확률은 HMM의 96.49%보다 0.07% 낮았지만, 처리 소요 시간을 약 53% 감소시켰다.

키워드 : 한국어 형태소 분석기, 은닉 마르코프 모델, 동형이의어, 태깅

1. 서 론

형태소 분석기는 다양한 필요성에 의해 많이 연구되어 왔으며, 근래 들어서는 의미(동형이의어) 분별에 대한 연구가 진행되고 있다. 용언은 동형이의어별로 문형구조가 다른 경우가 많아 구문구조 분석에 앞서 동형이의어가 분별된다면 구문구조 분석의 여러 중의성을 해소할 수 있다. 또한 동형이의어별로 기본 의미가 다르므로 기계번역 등에서 정확한 대역어(target word)를 제공할 수 있다. 예를 들어 "*그녀는 차를 타고 떠났다.*"의 문장에서 '차를'과 '타고'는 각각 '차/NNG+를/JKO', '타/VV+고/EC'로 동일한 형태소로 분석되지만 여러 동형이의어 중 하나로 해석될 수 있다.

표준국어대사전에 등재된 약 507,000 어휘 중 125,600 어휘(약 25%)가 동형이의어이며, 각 동형이의어는 의미번호가 정의되어 있다. 한국어의 의미처리를 위해서는 해당 어휘의 동형이의어가 분별되어야 하며, 이것은 해당하는 의미번호로 태깅되어야 한다는 뜻이다. 이러한 과정은 주변 문맥을 이용하여 가능하다. 위 예문에서 '차를'과 '타고'는 주변 문맥에 의해 '차_06/NNG+를/JKO'과 '타_02/VV+고/EC'로 태깅된다.

※ 본 논문은 2012년 2014년 정부(교육과학기술부)의 재원으로 한국연구재단 연구사업의 지원을 받아 수행된 연구임(No. 2012R1A1A2006906, 2014R1A1A2009506).
† 정 회 원 : 울산대학교 지능형컴퓨터연구실 연구교수
†† 종신회원 : 울산대학교 전기공학부 IT융합전공 교수
Manuscript Received : June 20, 2014
First Revision : August 14, 2014; Second Revision : September 3, 2014
Accepted : September 3, 2014
* Corresponding Author : Ock Cheol Young(okcy@ulsan.ac.kr)

2. 관련 연구

2.1 품사 태깅

초기의 형태소 분석 관련 연구들은 형태소를 분석하고 품사만 태깅하는 것에 집중하였으나, 품사 태깅의 정확률이 일정 수준 이상 오른 뒤에는 동형이의어 태깅 연구도 이루어졌다. 그리고 동형이의어 태깅 여부를 떠나서 대부분의 연구에서 HMM이 사용되었으며, 적지만 다른 알고리즘을 사용한 연구도 있다.

김진동(1997)은 한국어의 특성을 고려하여 어절의 첫과 끝에 위치한 품사만을 사용하여 4가지 전이 모델을 구성하고, 4가지 확률 값을 모두 곱하여 하나의 전이확률을 계산했다[1]. 전이확률을 적용하고 최적열을 구하는 과정에서 HMM을 사용하였다. 황명진(2007)은 앞 어절의 마지막 형태소와 함께 뒤 어절의 처음 혹은 끝 형태소와의 전이 링크만으로도 어절 간 전이확률 계산 시 필요한 대부분의 정보를 얻을 수 있고, 문맥에 따라 두 링크 중 하나만 필요하다는 관찰을 토대로 규칙을 이용해 두 전이링크 중 하나를 선택해 전이확률을 계산에 사용하는 “다이나믹 링크 모델”을 제안했다. 이 모델도 마찬가지로 HMM을 사용하였고 세종말뭉치 460만 어절을 학습해 실험한 결과 96.60%의 정확률을 보였다. 박희근(2007)은 어절별 중의성 해소 규칙과 trigram의 HMM을 이용하여 품사 태깅을 하였다[2]. 태깅 결과 97.93%의 정확률을 보였으나 실험 말뭉치는 1,000 문장의 17,384 어절로서 충분한 양의 데이터의 실험으로 보기는 어렵다.

영어에는 품사 태깅을 위해 HMM을 이용한 연구[3]와 규칙 기반 연구[4], 그리고 선형 분리 시스템을 이용한 연구[5] 등이 있다. 다양한 전이모델을 복합적으로 계산한 HMM은 다른 모델들과 충분히 비교할만한 높은 성능이 나왔다[3]. 이 외에 최근에는 조건부 랜덤 필드(Conditional Random Field, 이하 CRF)를 적용한 연구가 주목받고 있다[18]. 한국어 처리 분야에서도 CRF가 활발하게 연구되고 있으나 형태소 분석 및 태깅 부분의 최근 연구 결과 정확률은 약 95.21%로 HMM을 적용한 최근의 연구들에 비하여 낮은 정확률을 보여주었다[19]. 그러나 이 CRF 적용에 관한 연구는 아직 초기단계이며 개선의 여지가 많아 현재 활발하게 연구되고 있는 중이다.

2.2 동형이의어 분별

품사뿐만 아니라 의미번호를 구분하려는 시도도 있었다. 사용되는 학습 정보에 따라 연구들을 구분할 수 있으며, 영어권에서 대표적으로 초기의 원시 말뭉치를 사용한 연구[6]가 있으며, 한국어에서는 사전의 뜻풀이를 사용한 연구[7~11], 기분석 말뭉치를 사용한 연구[9, 12, 14, 15, 16], 그리고 어휘의미망을 사용한 연구[13, 14]가 있다. 이 중에 사전 뜻풀이와 소량의 말뭉치를 같이 사용한 연구가 있으며[9], 기분석 말뭉치와 어휘의미망을 같이 사용한 연구가 있다[14].

한국어에서 동형이의어 분별 연구는 대부분 일부 동형이의어에 한정하여 실험하였다. 임수종(1998)은 문맥에서 추출한 가중치 정보를 이용하여 한국어 동사의 의미 중의성을 해소하는 모델을 제안하였고 4개의 한국어 동사에 대해 84%의 정확률이 나왔다[7]. 이왕우(2004)는 유용한 구문 패턴을 바탕으로 사전 뜻풀이와 150만 어절의 말뭉치에서 어휘 공기 집합을 추출하여 동형이의어의 분별에 이용하였고 고빈도의 469개 동형이의어를 대상으로 실험 결과 92.23%의 정확률을 보였다.

신준철(2012)은 비교적 최근의 HMM 연구로, 약 천만 어절의 세종말뭉치를 사용하였다[15]. 우선 어절별로 분석 후보를 생성하기 위해서 기분석 부분어절사전을 활용한 한국어 형태소 분석기를 사용하였으며[17], 단계별 전이모델을 이용하였다. 약 990만 어절을 학습하고 110만 어절을 실험하였으며 어절 단위로 형태소, 품사, 동형이의어 번호가 모두 맞아야 정답으로 인정하였다. 그렇게 실험하여 정확률 96.49%를 보였다.

그리고 신준철(2013)은 HMM의 느린 속도를 해결하기 위해 간단하고 빠른 확률 모델도 제안하였다[16]. 현재 어절의 후보를 결정하기 위해 뒤 어절의 처음 2음절 또는 앞 어절의 마지막 2음절과의 조건부 확률을 계산하였다. 본 논문은 상술한 신준철(2013)의 연구를 보강한 것으로, 인접 어절의 부분 정보를 조건으로 현재 어절을 태깅하는 다양한 형태의 조건부 확률 모델과 단계별 적용 방법을 제안한다.

3. 부분어절 조건부확률 모델

일반적으로 형태소 분석 및 태깅은 크게 2단계로 나뉜다. 첫 단계에서는 어절을 분석하여 형태소와 품사가 표시되는 후보들을 생성하고, 두 번째 단계에서는 문맥정보(주로 인접한 어절)를 이용하여 최적의 후보를 선택하는데, 이를 태깅이라고 한다. 본 논문에서 제안하는 방법은 첫 단계에서 품사와 함께 동형이의어 번호가 표시되는 후보를 생성하고, 태깅 단계에서는 정확한 동형이의어가 표시된 후보를 선택하는 것이다. 이 태깅 단계에서 사용할 수 있는 방법으로는 HMM을 적용한 방법들이 이미 연구되어 있으며, 본 논문에서는 태깅 속도를 빠르게 하는 다른 방법을 제안한다.

HMM 기반의 태깅 과정은 인접한 어절이 가지는 모든 후보와의 확률을 계산하기 때문에 많은 시간이 소요된다. 만약 인접 어절의 후보 정보(은닉정보, 형태소 원형과 품사 및 동형이의어 정보)를 사용하지 않고 부분어절(관찰정보, 표층형)만을 이용해서 최적 후보를 결정할 수 있다면 속도를 향상시킬 수 있을 것이다. 본 논문은 인접 어절의 후보 보다 부분어절 정보를 우선적으로 사용하는 "부분어절 조건부확률(Sub-word Conditional Probability : SCP)"을 제안한다.

3.1 어절 정보 우선 사용

일반적으로 현재 어절의 의미(동형이의어)를 결정하기 위해서 인접 어절의 분석결과를 반드시 알 필요는 없다. 기분석 어절의 최고빈도 분석결과만을 이용하더라도 93.5%의 경우는 동형이의어 번호까지 정확히 태깅할 수 있으며[15], 나머지의 경우도 대부분 인접 어절의 표층형만으로 충분히 의미를 결정할 수 있다. 예를 들어서 "*사과를 먹었다.*"에서 '사과'의 의미를 결정하기 위해서 '먹었다.'의 분석 결과(형태소, 품사, 동형이의어 등)를 알 필요는 없으며, 단지 '먹었……'이라는 어절 정보만으로도 추측이 가능하다. 실제로 세종말뭉치에서 "*사과를 먹다.*"에 해당하는 어절쌍을 찾으면 '사과'의 의미는 한 가지로만 태깅되어 있다. 세종말뭉치에서 '사과를' 다음 어절이 '먹었' 또는 '먹는' 등으로 시작하는 경우를 찾으면 Table 1과 같다. 각 줄의 마지막 숫자는 빈도를 의미한다.

이와 같이 인접 어절의 처음 2개의 음절만 알더라도 현재 어절의 의미를 추측할 수 있음을 알 수 있다. 다른 예로 세

Table 1. Pairs of Eojeols About "eat an apple"

Pair of Eojeols	Tagging Result	Frequency
사과를 먹었……	사과_05/NNG+를/JKO	1
사과를 먹는……	사과_05/NNG+를/JKO	1
사과를 먹고……	사과_05/NNG+를/JKO	1
사과를 먹으……	사과_05/NNG+를/JKO	2
사과를 먹어……	사과_05/NNG+를/JKO	1
사과를 먹은……	사과_05/NNG+를/JKO	1

Table 2. Pairs of Eojeols for "ride a car"

Pair of Eojeols	Tagging Result	Frequence
차를 타고……	타_02/VV	120
차를 타고……	타_03/VV	1
차를 타면……	타_02/VV	5
차를 타게……	타_02/VV	1
차를 타려……	타_02/VV	4
차를 타는……	타_02/VV	4
…	…	…

Table 3. Pairs of Eojeols for "ride a car"

Tagging Result	Frequence
차_06/NNG+를/JKO 타_02/VV+고/EC	119
차_09/NNG+를/JKO 타_03/VV+고/EC	1
기차_01/NNG+를/JKO 타_02/VV+고/EC	57
자동차/NNG+를/JKO 타_02/VV+고/EC	23
승용차/NNG+를/JKO 타_02/VV+고/EC	26
전차__10/NNG+를/JKO 타__02/VV+고/EC	11
…	…

종말뭉치에서 "*차를 타면*" 또는 그와 비슷한 어절쌍에서 용언 '타/VV'가 어떤 동형이의어로 태깅되었는지를 찾으면 Table 2와 같다. 이들 중에서 예외적으로 "*차를 타고……*"만 2가지 경우로 나타났으나 총 121개 중에서 1개만이 다른 뜻으로 사용되고 있다. 만약 좌측 어절이 '차를'보다 긴 경우, 예를 들어 '자동차를'이라면 마지막 2음절 '차를'만 참조하는 방법으로도 태깅이 가능하다. 세종말뭉치에서 "……*차를 타고*……"에 해당하는 어절쌍은 총 389개이며 이 중에서 '타_02/VV'가 348번, '타_03/VV'는 오직 1번 나타났다. Table 3은 세종말뭉치에서 "……*차를 타고*……"에 해당하는 어절쌍들 중의 일부를 나타낸 것이다. 이런 예를 통해서 대부분의 경우에 인접 어절에서 2개의 음절만 참고하여도 동형이의어 태깅이 가능할 것임을 알 수 있다. 인접 어절 정보 외에 HMM을 사용해야 의미 태깅이 가능한 경우는 체인(chain)효과가 필요한 경우다. bigram 단위의 처리를 하더라도 HMM은 직접 인접하지 않은 어절로부터도 의미 결정에 영향을 받을 수 있기 때문에 체인 효과를 받을 수 있다. 예를 들어서 "*따뜻한 차를 따라라.*"에서 '따라라'의 의미를 결정하는 것이다.

차_06[car]/NNG+를/JKO	따르_01[follow]/VV+아라/EF
차_09[tea]/NNG+를/JKO	따르_02[pour]/VV+아라/EF

이 예에서는 '따라라'의 의미를 알기 위해서 반드시 '차'의 의미를 알아야 한다. 그리고 '차'의 의미는 '따뜻한' 어절을 통해 알 수 있으며 HMM을 사용하면 이런 경우를 해결할 수 있다. 다만 한국어에서 이런 경우는 매우 드물게 발생하며, 만약 멀리 떨어진 어절을 참조해야 하는 경우 HMM을 사용한다 하더라도 항상 해결된다는 보장은 없다. 따라서 본 논문은 대부분의 경우 직접 인접한 어절의 부분어절만으로 의미를 추측할 수 있다고 가정하고, 인접 어절 정보를 우선적으로 사용하는 SCP 모델을 제안한다.

3.2 부분어절 조건부확률 모델의 기본 형태

태깅이란 어절열(문장)과 후보들이 주어졌을 때 각 어절마다 가장 적절한 후보를 선택하는 것이다. 주어진 문장을 Sent라고 정의하고 i번째 어절은 w_i라고 정의한다. 태깅을

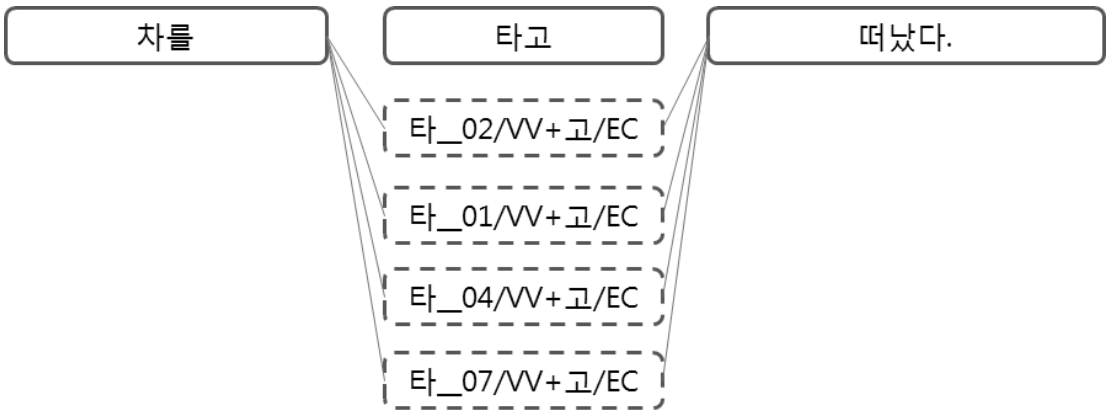

Fig. 1. An Example of Sub-word Conditional Model

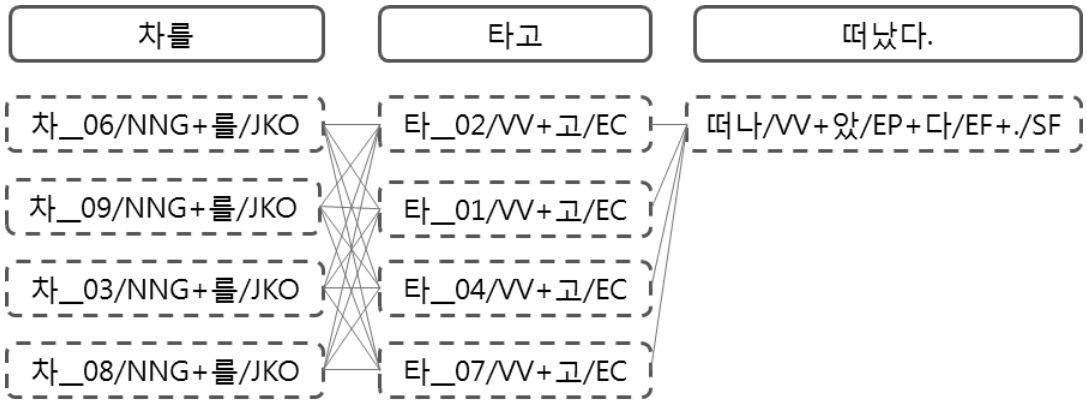

Fig. 2. An Example of HMM

시작하기 전에 각 어절은 이미 분석되어서 여러 개의 후보(형태소, 품사, 동형이의어 번호로 구성)를 가지고 있다고 가정하며 i번째 어절의 j번째 후보를 $t_{i,j}$라고 정의한다. a_i는 i번째 어절의 후보들 중 1개를 뜻한다. 그러면 하나의 어절 w_i에 대한 태깅을 식 (1)로 정의할 수 있다.

$$\begin{aligned} &Sent = w_1 w_2 \ldots w_n \\ &t_i = \{t_{i,1}, t_{i,2}, \ldots, t_{i,m}\} \\ &a_i \in t_i \\ &Tag(w_i) = \arg max_j P(t_{i,j} | w_1, \ldots w_n, a_1, \ldots a_{i-1}, a_{i+1} \ldots a_n) \end{aligned} \quad (1)$$

아무리 큰 말뭉치를 학습하더라도 어절열 전체를 조건부로 사용하는 식은 여러 이유에서 현실성이 없다. 재현율이 낮을 뿐만 아니라 많은 시간이 필요하기 때문이다. 이 문제를 해결하기 위해 마르코프 가정(직전 상태만이 현재 상태에 영향을 준다는 가정)을 도입하는 것이 HMM이다. SCP에서 사용하는 가정은 마르코프 가정과 유사하지만 두 가지 다른 점이 있다. 첫 번째로 SCP는 직전 상태(i-1) 뿐만 아니라 직후의 상태(i+1)도 현재 상태(i)에 직접적인 영향을 준다고 가정한다. 두 번째로 SCP는 직후와 직전 상태에서 오직 관찰된 상태(어절, 표층형, w_{i-1}, w_{i+1})만 현재 상태에 영향을 준다고 가정한다. 이 가정을 도입하면 SCP를 식 (2)와 같이 정의할 수 있다.

$$\begin{aligned} &Tag(w_i) = \arg max_j P(t_{i,j} | w_{i-1}, w_i, w_{i+1}) \\ &P(t_{i,j} | w_{i-1}, w_i, w_{i+1}) \cong P(t_{i,j} | w_{i-1}, w_i) \times P(t_{i,j} | w_i, w_{i+1}) \end{aligned} \quad (2)$$

예를 들어 Table 4와 같이 문장 Sent가 "*맛있는 사과를 먹었다.*"이면 n은 3이고 w_1은 첫 번째 어절 '맛있는'이 된다. 각 어절은 후보(형태소, 품사, 동형이의어가 표시된 것)들을 가지고 있으며 현재 태깅할 어절을 2번째 어절 '사과를'이라

Table 4. An Example Sentence and Candidates

w_1	w_2	w_3
맛있는 (delicious)	사과를	먹었다. (ate)
	$t_{2,1}$=사과_05(apple)/NNG+를/JKO	
	$t_{2,2}$=사과_08(apology)/NNG+를/JKO	

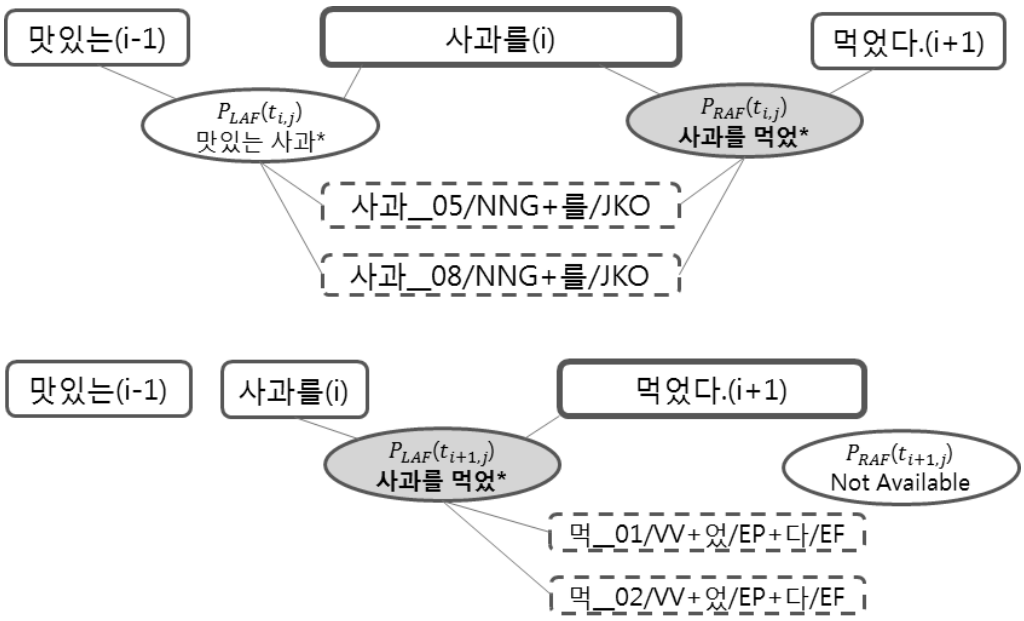

Fig. 3. Two Conditional Probabilities With Same Condition

고 가정하면 i는 2가 된다. '사과를'은 대표적으로 2개의 분석 후보를 가지고 있기 때문에 $T_{2,1}$은 '사과_05[apple]/NNG+를/JKO'이고 $T_{2,2}$는 '사과_08[apology]/NNG+를/JKO'이라고 가정한다.

식 (2)는 '사과를'의 각 후보에 대한 확률을 계산하기 위해 "*맛있는 사과를*"을 조건부로 계산한 확률값과 "*사과를 먹었다.*"를 조건부로 계산한 확률값을 곱한다. SCP는 이 값이 최대인 후보를 정답으로 선택한다.

HMM과는 다르게 SCP는 인접한 어절 정보만 사용하기 때문에 계산해야 할 경우의 수가 적다. Fig. 1은 SCP에 필요한 계산들을 선으로 표현한 것으로 "*차를 타고 떠났다.*"에서 '타'에 의미를 태깅하는 예이다. HMM으로 같은 예를 처리하기 위해서는 더 많은 계산이 필요하며 이것을 표현한 것이 Fig. 2이다. 이 둘을 비교해보면 SCP가 HMM보다 계산량이 적음을 시각적으로 알 수 있다. Fig. 1은 후보 하나당 좌우로 2개의 선이 존재하지만, Fig. 2는 인접한 어절의 후보 수만큼 존재한다.

3.3 부분어절 조건부확률 모델의 단계별 적용

HMM의 단계별 적용[15]과 마찬가지로 SCP도 단계별로 적용하는 것이 가능하며 재현율과 정확률 측면에서 더 좋다. HMM의 단계별 적용에서는 어절의 처음 또는 끝 형태소 정보를 사용하며, SCP는 마지막 음절 또는 첫음절을 사용하는 방식으로 응용할 수 있다. 단 1음절 정보만을 사용하면 지나치게 정보량이 적어지기 때문에 처음 또는 마지막의 2음절을 사용하도록 한다. 현재(i) 어절의 전체와 우측(i+1) 어절의 처음 2음절을 사용한 확률 모델(using Right eojeol with All-First form : RAF)은 다음 식 (3)으로 정의한다. $s_{i,x}$는 i어절의 x번째 음절을 의미한다.

$$P_{RAF}(t_{i,j}) = P(t_{i,j}|w_i, s_{i+1,1}, s_{i+1,2}) \quad (3)$$

예를 들어서 "*맛있는 사과를 먹었다.*"에서 '사과를'의 의미가 '사과_05/NNG+를/JKO'로 결정될 확률을 RAF로 계산한다면, 말뭉치에서 현재 어절이 '사과를'이고 우측 어절이 '먹었'으로 시작할 때, 현재 어절의 분석이 '사과_05/NNG+를/JKO'로 분석될 확률을 계산한다. 따라서 학습 말뭉치에서 "*사과를 먹었다.*"뿐만 아니라 "*사과를 먹었지만*", "*사과를 먹었으나*", "*사과를 먹었겠지*" 등의 어절쌍들이 학습되어도 확률에 적용된다. 본 논문은 이런 어절쌍 조건을 "사과를 먹었*"으로 표기한다.

SCP는 3.2절의 가정에 따라 우측 어절(i+1)뿐만 아니라 좌측 어절(i-1)도 현재 어절(i)의 태깅에 직접적인 영향을 준다. RAF는 i+1어절을 참조할 뿐이며, i-1어절을 참조하기 위한 확률 모델(using Left eojeol with All-First form : LAF)을 구성할 수 있다. LAF는 i-1어절의 전체와 i어절의 첫 2음절을 참조하게 되며, P_{LAF}는 다음 식 (4)로 정의한다. 여기서 $m_{i,j,k}$는 i어절 j후보의 k번째 형태소(품사, 동형이의어 포함)를 의미하고 w_{LAF}는 LAF의 가중치를 의미한다.

$$P_{LAF}(t_{i,j}) = P(m_{i,j,1}|w_{i-1}, s_{i,1}, s_{i,2})^{w_{LAF}} \times P(t_{i,j}|w_i) \quad (4)$$

이렇게 RAF와 LAF를 구성하면, i어절 다음에 i+1어절을 태깅할 때 LAF의 조건은 i어절을 태깅하기 위한 RAF의 조건과 정확히 일치한다. Fig. 3은 P_{RAF}와 P_{LAF}의 조건이 일치하는 과정을 자세하게 보여준다. 이 특징을 이용하면 실제 완성된 시스템에서 두 확률 계산을 위해 필요한 말뭉치 학습자료를 같이 저장하고 검색할 수 있기 때문에 컴퓨터의 연산량과 학습자료의 크기를 많이 줄일 수 있다.

LAF는 태깅 대상 어절(i)에 대해서 처음 두 음절만 조건에 포함하기 때문에 첫 번째 형태소에 대해서만 확률을 계산한다. 예를 들어, "*맛있는 사과를……*"에서 조건 "*맛있는 사과**"에는 현재 어절 '사과를'에서 '를'이 제외되고 오직 '사과'만 조건에 포함되기 때문에 첫 번째 형태소 $m_{i,j,1}$(예: 사과_05/NNG)에 대해서만 확률을 고려한다. 이 때문에 나머지 형태소들에 대한 특성('를'이 '를/JKO'가 되게 하는 것)이 전혀 확률에 고려되지 못하는 단점이 있으며, 이를 보완하기 위해 현재 어절 전체를 조건으로 하는 확률 $P(t_{i,j}|w_i)$를 추가로 곱한다.

태깅하려는 어절이 세종말뭉치에서 등장한 적이 한 번도 없다면 $P(t_{i,j}|w_i)$는 직접적으로 계산이 불가능하지만 후보 생성 단계에서 발생하는 후보 점수를 빈도로 취급하여 계산할 수 있다[17]. 다음은 '사과인사를' 어절의 후보들과 점수이다.

- 사과_05/NNG+인사_02/NNG+를/JKO(apple+bow) 100
- 사과_08/NNG+인사_02/NNG+를/JKO(apology+bow) 82
- 사과_08/NNG+인사_01/NNG+를/JKO(apology+personage) 8

각 줄의 우측에 있는 숫자는 해당 후보의 점수를 의미하며, 점수계산은 어절 내에서 품사 또는 형태소의 전이확률이나, 각 형태소(사과_05/NNG, 를/JKO 등 품사와 동형이의어 번호를 포함한 형태)의 빈도 등을 사용하여 계산된다[17]. 위의 점수를 후보에 대한 빈도와 같은 의미로 취급하면 $P(t_{i,j}|w_i)$는 다음과 같이 계산할 수 있다.

$$P(t_{i,j}|w_i) \cong \frac{Score(t_{i,j})}{\sum_{a=1}^{m} Score(t_{i,a})} \quad (5)$$

예를 들어, "*진심으로 사과인사를 드립니다.*"를 처리하기 위해 LAF는 현재 처리 중인 어절 '사과인사를'의 좌측 어절인 '진심으로' 어절 정보를 이용하여 '사과'의 의미가 apology가 되게 하고, 후보 점수를 이용해 '인사'가 bow가 되도록 유도할 수 있다.

SCP의 기본 형태는 식 (2)와 같이 좌측 어절을 조건으로 한 확률과 우측 어절을 조건으로 한 확률의 곱을 계산한다. LAF와 RAF모델을 사용할 경우에는 식 (6)처럼 P_{LAF}와 P_{RAF}의 곱을 계산한다. 이때 LAF의 가중치인 w_{LAF}는 RAF에 비하여 LAF가 얼마나 더 중요한지를 표현하게 된다.

$$P_{SCP}(t_{i,j}) = P_{LAF}(t_{i,j}) \times P_{RAF}(t_{i,j}) \quad (6)$$

예를 들어서 "*따뜻한 차를 따라라.*"에서 '차를' 어절의 의미를 결정하기 위해서 '따뜻한' 어절 정보를 이용하는 것이 LAF이며, '따라라.'를 이용하는 것이 RAF이다. 여기서 w_{LAF}는 '차를' 어절의 의미 결정에서 '따라라.'보다 '따뜻한'이 얼마나 더 중요한지를 표현한다고 볼 수 있다. w_{LAF}값은 반복된 실험을 거쳐 최적의 값을 선택하는 것이 가장 좋다.

Fig. 3으로 설명하였듯이 LAF와 RAF는 같은 어절쌍에 대해서 같은 조건을 가진다. 다음은 세종말뭉치를 대상으로 LAF와 RAF의 어절쌍 조건 "*그 사과**"를 만족하는 어절들의 분석과 빈도다. '사과*'에 해당하는 우측 어절은 첫 번째 형태소만 표시하였다.

그_01/MM : 9	사과_05/NNG : 7 사과_08/NNG : 1 사과나무/NNG : 1

세종말뭉치에서 어절쌍 조건 "*그 사과**"를 만족하는 경우는 총 9개로, 좌측은 '그_01/MM'으로만 분석되며 우측은 3가지 정보가 존재하지만 '사과_05[apple]'가 가장 많다. 우측 어절은 처음 2글자만 '사과'이면 조건을 만족하기 때문에 2음절이 넘는 형태소 '사과나무'도 존재한다.

LAF와 RAF도 학습량 부족 현상으로 적용이 불가능할 수 있다. 예를 들어서 "*맛있는 사과라도*"를 태깅한다고 가정했을 때 세종말뭉치에서 확률조건 "맛있는 *사과**"에 해당하는 경우가 하나도 없다면 처리가 불가능하다. 이를 재현실패라고 하며, 재현에 실패한 경우에는 재현율이 더 높은 대체모델이 필요하다. 단계별 적용이란 기존모델이 재현실패한 경우에 대체모델을 사용하는 것을 의미한다. 본 논문에서 제안하는 RAF의 대체모델은 현재 어절의 마지막 두 음절과, 우측 어절의 처음 두 음절 정보를 사용하는 모델이다(using Right eojeol with End-First form : REF). 식 (7)은 REF의 확률을 정의한 것이며, 여기서 L은 해당 어절 후보의 마지막 형태소를 의미하고 N은 마지막 음절을 의미한다. 그리고 w_{REF}는 REF의 가중치이다.

$$P_{REF}(t_{i,j}) = P(m_{i,j,L}|s_{i,N-1}, s_{i,N}, s_{i+1,1}, s_{i+1,2})^{w_{REF}} \times P(t_{i,j}|w_i) \quad (7)$$

$$P_{LEF}(t_{i,j}) = P(m_{i,j,1}|s_{i-1,N-1}, s_{i-1,N}, s_{i,1}, s_{i,2})^{w_{LEF}} \times P(t_{i,j}|w_i) \quad (8)$$

$$P_{SCP}(t_{i,j}) = P_{LSCP}(t_{i,j}) \times P_{RSCP}(t_{i,j})$$

$$LSCP = \begin{cases} LAF\ if\ it\ is\ available \\ LEF\ \ if\ \ LAF\ failed \end{cases} \quad (9)$$

$$RSCP = \begin{cases} RAF\ if\ it\ is\ available \\ REF\ if\ \ RAF\ failed \end{cases}$$

LAF의 대체모델은 현재 어절의 처음 두 음절과 좌측 어절의 마지막 두 음절 정보를 사용한다(using Left eojeol with End-First form : LEF). 식 (8)은 LAF의 확률을 정의한 것이고 w_{LEF}는 LEF의 가중치이다.

좌측 어절을 조건으로 사용한 모델을 통틀어 Left-SCP(이하 LSCP)라고 하며 LAF와 LEF가 이에 속한다. 우측 어절의 경우는 Right-SCP(이하 RSCP)라고 하고 RAF와 REF가 이에 속한다. SCP 단계별 적용 모델의 전체 식은 다음과 같이 표현할 수 있다.

최상의 경우에 하나의 어절(i)을 태깅하기 위해 LSCP에서는 LAF가 선택되고, RSCP에서는 RAF가 선택된다. 이 경우에 RAF의 가중치 w_{RAF}는 RAF가 LAF에 비해 얼마나 더 중요한지를 표현한다. 만약 RAF 대신에 REF가 선택된다면 REF의 가중치 w_{REF}는 LAF에 비해 REF가 얼마나 더 중요한지를 표현한다. 따라서 w_{RAF}와 w_{REF}는 모두 LAF를 기준으로 정해지는 값으로 이해할 수 있으며 서로 비교가 가능하다. 이 원리를 모든 가중치(w_{RAF}, w_{REF}, w_{LEF})에 적용하면, 각 가중치들은 해당 모델이 다른 모델들에 비해 얼마나 더 중요한지를 표현하는 것이 된다.

예를 들어, "*맛있는 사과라도*"를 태깅하기 위한 REF와 LEF의 어절쌍 조건은 "**있는 사과**"이다. 세종말뭉치에서 이 조건에 해당하는 어절쌍을 찾아서 좌측과 우측 분석을 나열하면 다음과 같으며, 이 정보를 기반으로 태깅하면 '사과_05[apple]'가 선택될 것이다.

3.4 용언 예외처리

3.3절의 모델들은 확률조건으로 형태소 원형을 사용하지 않고 어절의 표층형 정보만을 사용하기 때문에 음운변동이 잦고 다양한 어미와 조합이 가능한 용언을 처리해야 하는 경우에 재현율이 다소 낮을 것으로 예상된다. 예를 들어, "*사과를 먹자면*"은 RAF와 LAF로 태깅이 불가능하다. 세종말뭉치에 "*사과를 먹자**" 조건을 만족하는 어절쌍이 없기 때문이다. 그러나 '사과를' 어절 다음에 '먹_02[eat]/VV'동사가 나타난 경우는 여럿 존재한다. 단지 표층형이 '먹자*' 형태로 나타난 것이 없을 뿐이다. HMM을 응용하여 어절의 첫 번째 형태소 원형을 조건으로 사용한다면 이런 경우에도 성공적으로 분석을 수행한다. 이런 특징 때문에 SCP는 HMM보다 재현율이 낮아질 수 있다.

따라서 우측 어절이 용언일 경우에만 어절 정보가 아닌, 어간(세종 품사 태그가 VV 또는 VA)을 확률의 조건으로 사용하는 방법도 생각할 수 있다. 우선적으로 어절 정보를 사용해보고, 이에 실패할 경우에 차선책으로 어간 정보를 사용한다. 하나의 어절은 여러 개의 후보를 가질 수 있으며, 각 후보들의 첫 번째 형태소가 어간일 때 그 형태소들은 서로 형태가 다를 수 있다. 예를 들어서 '간다'의 후보는 다음과 같다.

가_01(go, move)/VV+ㄴ다/EF
가_01(ongoing)/VX+ㄴ다/EF
갈_01(replace)/VV+ㄴ다/EF
갈_02(grind)/VV+ㄴ다/EF

후보는 총 4가지이지만 동형이의어 번호와 품사를 제거하면, 어간은 '가'와 '갈' 2가지로 줄어든다. 이와 같이 어간이 여러 가지일 경우에는 각각에 대해서 확률을 계산하고, 그 중에서 가장 높은 확률을 사용한다. 즉, 위와 같이 '가'와 '갈' 2가지가 존재하면 확률을 2번 계산한다. 이렇게 되면 SCP모델은 인접 어절의 은닉상태를 참조하는 것이 되지만 은닉상태 중에서 오직 어간 정보(품사와 의미번호를 제외한 것)만 사용한다는 특징이 있다. 한국어에서 하나의 용언 어절이 가질 수 있는 어간의 최대 가짓수는 매우 제한적이며 대부분은 1개이다. 세종말뭉치에 포함된 어절 중에서 한 어절이 가지는 어간은 최대 4가지이고, 그런 어절은 '그러면', '날', '난'뿐이다. 그리고 세종말뭉치에서 용언 어간을 가진

는/ETM : 8	사과_05/NNG : 7 사과_08/NNG : 1 사과나무/NNG : 1

어절은 전체에서 31%이며, 용언 예외처리는 이전 과정(RAF, LAF)에 실패할 경우에 시도되기 때문에 실제 적용 비율은 매우 낮을 것으로 추측된다. 만약 SCP의 시간복잡도를 계산하기 위해 최악의 경우를 가정한다면 어간의 가짓수는 최대가 4이기 때문에 시간복잡도에서 생략되고, SCP의 시간복잡도는 $O(n \times m)$가 된다.

$v_{i,l}$을 i어절 l번째 어간 정보(예: '간다'에서 '가' 또는 '갈')라고 정의할 때, 다음 식 RAF2는 RAF에 용언 예외처리를 적용한 것이다. 예를 들어, "*사과를 먹었다.*"에서 '사과_05/NNG+를/JKO'가 선택될 확률은, 우측 어절의 어간이 '먹'인 것을 조건으로 계산된다.

$$P_{RAF2}(t_{i,j}) = \max_l(P(t_{i,j}|w_i, v_{i+1,l})) \tag{10}$$

같은 방법으로 LAF2, REF2, LEF2도 만들 수 있으며 이것이 식 (11)이다. 종합하면 RSCP에는 RAF, RAF2, REF, REF2가 차례대로 포함되며, 순서대로 시도된다. 그리고 LSCP에는 LAF, LAF2, LEF, LEF2가 포함되며 순서대로 시도된다. "*사과를 먹자면*"은 RAF, LAF로 분석할 수 없기 때문에 RAF2, LAF2를 시도하게 된다. 이를 위해서는 우선 우측 어절 '먹자면'을 분석해서 용언이 나타나는지 확인해야 하며, 다음은 '먹자면'을 분석하여 나타난 후보들이다.

먹_02[배_속에_들여보내다, eat]/VV+자면/EC
먹_02[배_속에_들여보내다, eat]/VV+자면/EF
먹_01[하지_못하게_되다, lose ability]/VV+자면/EC
먹_02[행동을_강조하는_말, emphasize]/VX+자면/EC
먹_02[행동을_강조하는_말, emphasize]/VX+자면/EF

사과_05/NNG+를/JKO : 7	먹_02/VV : 7

후보는 5가지로 나타나지만 품사와 동형이의어 번호를 제거하면 어간은 모두 '먹'이기 때문에 RAF2, LAF2의 확률조건은 "*사과를 먹(V)**" 하나가 된다. 다음은 세종말뭉치에서 이 조건에 만족하는 분석 정보다. 이 정보를 이용하면 "*사과를 먹자면*"을 정확하게 태깅할 수 있다.

3.5 최소빈도

P_{SCP}는 현재 어절의 우측 어절을 조건으로 하는 확률 P_{RSCP}와, 좌측 어절을 조건으로 하는 확률 P_{LSCP}의 곱으로 계산되기 때문에 간혹 모든 후보가 확률 0으로 계산될 가능성이 있다. 예를 들어, Fig. 4에서 '차를'의 의미를 결정하기 위해서 "*맛있는 차를*" 어절쌍에 LAF가 적용되었고, 후보 중에서 '차_09/NNG'를 포함하는 후보만 학습(빈도 20)되어 있음을 발견하였다. 그리고 "*차를 사면*" 어절쌍에서는 RAF가 적용되었고 '차_06/NNG+를/JKO' 후보에서 빈도 10으로 학

Fig. 4. An Example Sentence that Every Candidate's Probability is 0

$$\begin{aligned}
P_{LAF2}(t_{i,j}) &= \max_l(P(m_{i,j,1}|w_{i-1}, v_{i,l}))^{w_{LAF}} \times P(t_{i,j}|w_i) \\
P_{REF2}(t_{i,j}) &= \max_l(P(m_{i,j,L}|s_{i,N-1}, s_{i,N}, v_{i+1,l}))^{w_{REF}} \times P(t_{i,j}|w_i) \\
P_{LEF2}(t_{i,j}) &= \max_l(P(m_{i,j,1}|s_{i-1,N-1}, s_{i-1,N}, v_{i,l}))^{w_{LEF}} \times P(t_{i,j}|w_i)
\end{aligned} \tag{11}$$

$$\begin{aligned}
\arg max_j P_{RAF}(t_{i,j}) &= arg max_j P(t_{i,j}|w_i, s_{i+1,1}, s_{i+1,2}) \\
&= \arg max_j \frac{Freq(t_{i,j}, w_i, s_{i+1,1}, s_{i+1,2})}{\sum_{a=1}^{m} Freq(t_{i,a}, w_i, s_{i+1,1}, s_{i+1,2})} \\
&= \arg max_j Freq(t_{i,j}, w_i, s_{i+1,1}, s_{i+1,2}) \\
&\cong \arg max_j SFreq(t_{i,j}, w_i, s_{i+1,1}, s_{i+1,2})
\end{aligned} \tag{12}$$

$$SFreq(x) = \begin{cases} 0.1 & if\ \ Freq(x)\ \ is\ \ 0 \\ Freq(x) & otherwise \end{cases}$$

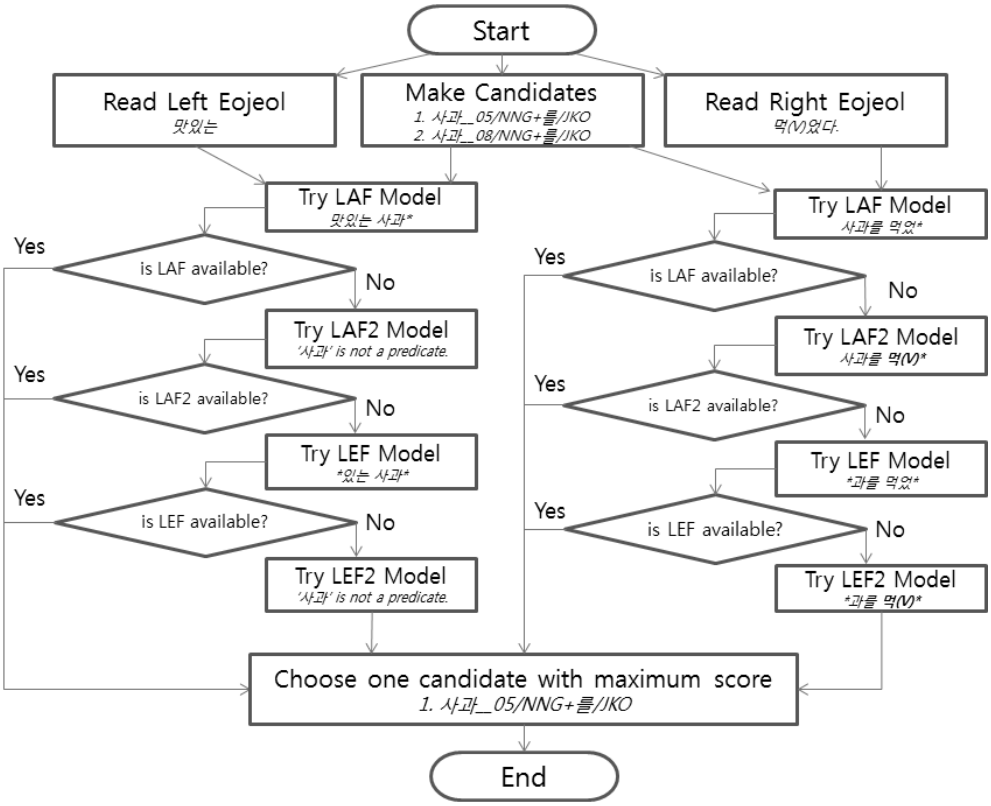

Fig. 5. Flow Chart of Sub-word Conditional Probability Model

$$\begin{aligned}\arg max_j P_{LAF}(t_{i,j}) &= \arg max_j \frac{P(m_{i,j,1}, w_{i-1}, s_{i,1}, s_{i,2})^{w_{LAF}}}{P(w_{i-1}, s_{i,1}, s_{i,2})} \times \frac{Score(t_{i,j})}{\sum_{a=1}^{m} Score(t_{i,a})} \\ &= \arg max_j SFreq(m_{i,j,1}, w_{i-1}, s_{i,1}, s_{i,2})^{w_{LAF}} \times Score(t_{i,j})\end{aligned} \tag{13}$$

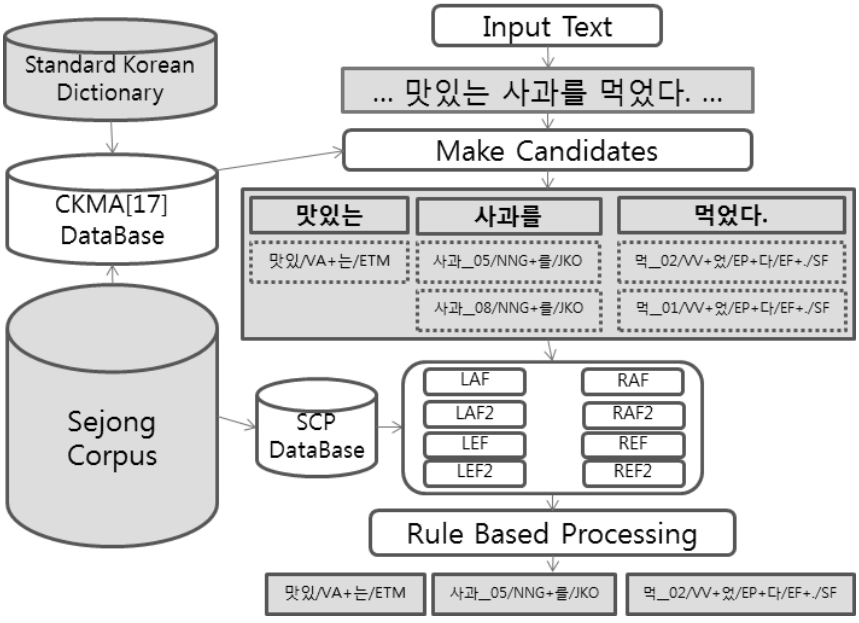

Fig. 6. The Entire System Block Diagram

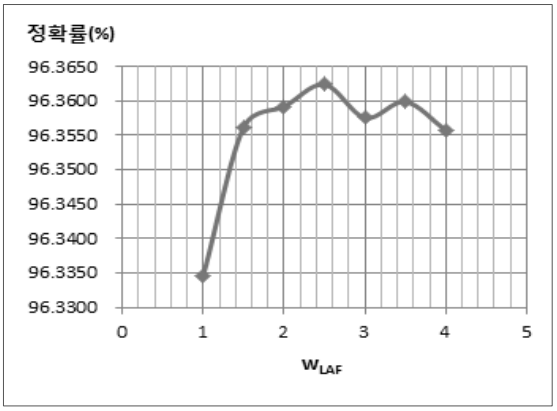

Fig. 7. Weight of "Left-All-First" and Accuracy

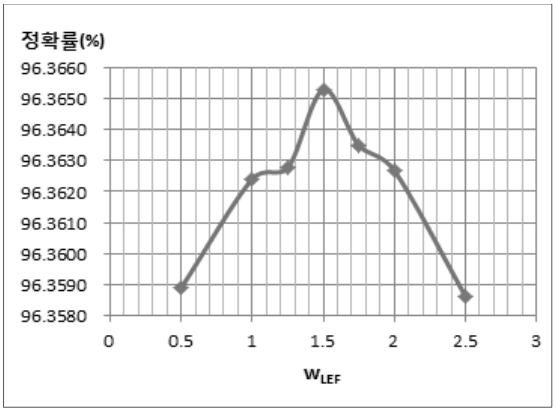

Fig. 8. Weight of "Left-End-First" and Accuracy

Table 5. Sub-word Conditional Model Weights

Name	Description	Value
w_{LAF}	All syllable of left eojeol(i-1) and first part of right eojeol(i).	2.5
w_{LEF}	Last two syllable of left eojeol(i-1) and first part of right eojeol(i).	1.5
w_{REF}	Last two syllable of left eojeol(i) and first part of right eojeol(i+1).	0.6

Table 6. Minimum Frequencies of Sub-word Conditional Probability Model

Name	Description	Value
RAF, RAF2	All syllable of left eojeol(i) and first part of right eojeol(i+1).	0.001
LAF, LAF2	All syllable of left eojeol(i-1) and first part of right eojeol(i).	0.01
REF, REF2, LEF, LEF2	Last part of left eojeol and first part of right eojeol.	0.1

Table 7. Results of 10 divided corpus cross-validation experiment

Corpus Part Number	1	2	3	4	5	6	7	8	9	10	AVG
accuracy (%)	96.199	96.230	96.206	96.221	96.202	96.206	96.231	96.211	96.235	96.169	96.211

습되어 있다. 이런 경우에 모든 후보의 P_{SCP}는 0이 된다.

이런 현상을 해결하기 위해서는 만약 빈도가 0으로 계산될 상황이더라도 실제로 적용되는 빈도를 0보다 아주 조금 높게 설정하는 방법이 있다. 예를 들어서 Fig. 4에서 0 대신에 0.1을 적용한다면 확률계산을 통해서 '차_09/NNG'후보가 선택될 것이다. 이렇게 적용되는 빈도를 최소빈도라고 정의하며 0.1처럼 0과 1 사이의 실수 값을 가진다.

식 (12)는 최소빈도의 적용 방법을 설명하기 위해 RAF를 빈도 기반의 식으로 표현한 것이다. Freq는 학습말뭉치에서 특정 조건을 만족하는 어절의 빈도를 의미하고 SFreq는 최소빈도가 적용된 빈도함수이다. argmax를 사용하는 문제 영역에서 공통된 값(식에서 분모 부분)은 생략할 수 있다. 식 (12)에서 최소빈도는 0.1로 가정하고 있으며, 빈도가 0으로 계산될 상황에 처하면 0 대신에 0.1이 적용된다. 예를 들어 Fig. 4에서 "*차를 사면*"에 적용되는 RAF값은 첫 번째 후보(차_06)에 대해서 10이고 두 번째 후보(차_09)에 대해서 0.1이 된다. 이 최소빈도 값은 실험을 반복하여 적절한 값을 선택한다.

3.6 전체 시스템

Fig. 5는 SCP의 모든 모델이 적용되는 순서도이다. 그림 상에 모든 내용은 "*맛있는 사과를 먹었다.*"에서 '사과를'의 후보를 선택하는 과정이며, 좌측 어절과 우측 어절 정보의 전체 또는 일부를 이용한다. 그림에서는 생략되어 있으나 SCP 단계별 모델의 마지막에는 어절쌍에서 좌측 어절의 첫 2음절과 우측 어절의 어간 정보를 확률조건으로 이용한 모

Table 8. Homograph Accuracy

Target	Name	Number of Eojeols	Ratio(%)
All Eojeol of Test-Set	T	1,108,204	
Homograph Eojeols of T	H	668,561	
Ratio of Homograph Eojeols	H/T		60.33
Eojeols with right Tag and right Morpheme in H	M	650,196	
Eojeols with right Homograph Number in M	R	640,456	
Accuracy of Homograph	R/M		98.50
Accuracy of Morpheme+Tag+Homograph	R/H		95.80
Accuracy of Morpheme+Tag	M/H		97.25

델이 시도된다. 그리고 부사어(그, 또, 늘, 많이 등)가 인접한 경우에는 부사가 아닌 가장 인접한 어절을 이용하여 확률계산을 한다. 예를 들어, "*사과를 또 먹었다.*"에서 '사과를'은 '먹었다.'와 인접한 것처럼 처리한다.

Fig. 6은 전체 시스템을 그린 것으로, SCP 모듈과 CKMA[17] 그리고 규칙 기반의 후처리 모듈을 담고 있다. 우선 세종말뭉치와 표준국어대사전을 이용해 CKMA를 학습시키고 이것으로 후보들을 생성한다. 그리고 세종말뭉치로 SCP 단계별 모델을 학습시킨 뒤에 태깅하면 각 어절마다 하나의 후보가 결정된다.

3.7 상수 결정

실제로 시스템을 구현할 때에는 가장 높은 확률값을 선택하는 문제에서 같은 범위 안(어절 i의 후보들)에서 공통되는 값은 계산할 필요가 없기 때문에 제거하였다. 예를 들어서 식 (12)가 있으며, LAF의 경우 식 (13)으로 변경할 수 있다. 이와 같이 변경하여 사용하면 가중치 w_{LAF}가 높을수록 후보 결정에서 LAF의 영향력이 커지게 된다.

SCP에서 사용되는 상수들은 수작업으로 결정하였다. 실험자의 한국어 지식을 바탕으로 초기값을 정하고 반복적인 실험을 통하여 정확률이 높게 나오는 값으로 결정하였다. Fig. 7과 Fig. 8은 상수들 중 대표적인 w_{LAF}, w_{LEF}의 결정 과정을 보여주고 Table 5와 Table 6은 모든 상수값을 나타내고 있다.

w_{LAF}, w_{LEF}, w_{REF} 중에서 가장 높은 것은 w_{LAF}였으며, 이것은 LAF가 가중치를 사용하는 모델 중에서 가장 중요한 정보라는 의미다. 그다음으로 w_{LEF}, w_{REF} 순으로 높았다. 최소빈도는 AF류가 EF류보다 많이 낮았으며, 그 원인은 AF류에서 빈도가 0인 후보는 정답이 아닐 확률이 높다는 의미로 분석된다.

4. 실험 및 평가

4.1 학습 말뭉치 10분할 교차 검증 실험

본 논문이 제안하는 SCP가 순수하게 말뭉치만을 학습하였을 때의 정확률을 측정하기 위해 표준국어대사전을 학습에서 제외하고 세종말뭉치만으로 실험하였다. 세종말뭉치를 10문장 단위로 분리한 뒤에 9문장씩을 학습하고, 나머지 1문장을 실험 데이터로 선택하였다. 그 실험 결과가 Table 7이다. 실험 번호 1은 10문장 중에 첫 번째 문장을 실험에 사용했다는 의미다. 정확률은 어절 단위로 측정하였으며, 형태소와 품사 그리고 동형이의어 번호가 모두 맞아야 정답으로 인정하였다.

Table 9. HMM and Sub-word Conditional Probability Model's Accuracy and Processing Time

Algorithm	Accuracy	Tagging Time
CKMA[17]	93.56%	0.0초
CKMA+HMM[15]	96.49%	21.1초
CKMA+SCP	96.42%	10.0초

Table 10. Error Examples of Sub-word Conditional Probability Model

Eojeol	Right Tagging Result	SCP Tagging Result
수학여행	수학여행/NNG	수학_05/NNG+여행_02/NNG
프랑스어에도	프랑스어/NNP+에/JKB+도/JX	프랑스_02/NNP+어_08/XSN+에/JKB+도/JX
사업장에서도	사업장/NNG+에서/JKB+도/JX	사업_04/NNG+장_45/XSN+에서/JKB+도/JX
공표하였다.	공표하/VV+었/EP+다/EF+./SF	공표_02/NNG+하/XSV+었/EP+다/EF+./SF

10묶음 교차 검증 실험 결과 정확률은 백분율에서 소수점 2번째 자리 수준에서 차이를 보였다. 이 차이는 매우 미미한 것으로 본 논문이 제안하는 SCP가 학습량만 동일하다면 정확률이 안정적으로 나타나는 것을 의미한다.

4.2 동형이의어 정확률

실험 데이터의 어절 중에 동형이의어를 포함하는 어절만을 대상으로 정확률을 측정해보았다. 실험을 위하여 세종말뭉치를 10문장 단위로 나눈 후에 한 단위마다 처음 9문장(총 990만 어절)을 학습하고 마지막 1문장(총 110만 어절)을 실험하였다. 그 결과가 Table 8에 나타나있다.

실험 데이터 총 110만 어절(T)에서 동형이의어를 포함한 어절(H)은 약 60%이며, 이 중에 형태소와 품사 분석이 정확한 어절(M, 동형이의어 분석이 틀린 것도 포함)은 97.25%였다. M 중에서 동형이의어 번호가 정답인 어절(R)의 비율은 98.50%로 매우 높은 정확률이 나왔다. 동형이의어를 포함한 어절(H)에서 형태소와 품사 그리고 동형이의어 모두를 맞출 확률은 95.80%로 모든 실험 데이터 어절을 대상으로 한 정확률 96.211%(Table 7 참조)보다 조금 낮았다.

4.3 성능 비교 및 오류 유형 분석

HMM 단계별 모델과 본 논문이 제안하는 SCP의 성능을 비교하기 위해 세종말뭉치 1,100만 어절과 표준국어대사전을 사용하였고, 알고리즘을 제외한 모든 환경은 동일하게 하였다. 표준국어대사전은 모두 학습하였고, 세종말뭉치를 10문장 단위로 나누었다. 그리고 10문장마다 마지막 1문장은 실험 데이터(110만 어절)로 분류하고 처음 1~9번 문장(990만 어절)은 학습하였다. 미등록 어절의 분석 후보를 생성하기 위해서는 CKMA[17]알고리즘을 사용하였다.

실험 결과가 Table 9에 나타나있으며, 비교해보면 HMM보다 SCP의 정확률은 0.07% 낮았지만 태깅 시간은 절반 이하로 줄었다. 문맥 정보를 활용하지 않고 오직 해당 어절만을 분석하여 생성된 후보 중에서 가장 점수(또는 빈도)가 높은 후보만을 선택했을 경우에는 93.56%의 정확률을 보였다. 이 경우는 후보 생성과정에서만 시간을 소요한 것이기 때문에 태깅 시간은 없는 것(0초)으로 간주한다.

HMM에서는 정확히 태깅되었으나 SCP에서 오류를 보인 어절의 예가 Table 10에 나타나있다. SCP는 주로 3음절이 넘는 형태소에서 HMM보다 많은 오류를 일으켰다. 이것은 SCP가 어절의 처음 또는 끝 2음절만을 사용하는 경우가 많기 때문인 것으로 분석된다. 형태소 '수학'은 여러 의미를 가지고 있기 때문에 동형이의어 번호가 표기되었으나 '수학여행'은 오직 하나의 의미를 가지기 때문에 동형이의어 번호가 존재하지 않는다. 같은 원리로 '프랑스어', '사업장', '공표하'는 동형이의어 번호가 없다.

'프랑스어'를 2개의 형태소 '프랑스_02/NNP+어_08/XSN'로 분석하는 경우도 기준에 따라서 정답으로 인정될 수 있으나, 본 논문에서 실험한 말뭉치에서는 '프랑스어'를 하나의 형태소 '프랑스어/NNP'로 분석하는 것을 기준으로 하고 있으므로 오답으로 처리하였다. 본 논문에서 실험한 말뭉치는 표준국어대사전에 등재된 단어를 가급적이면 하나의 형태소로 분석하는 것을 기준으로 한다. 그 외의 예들(수학여행, 사업장, 공표하)도 2개의 형태소로 분석하는 것이 반드시 틀린 것은 아니지만 본 논문의 기준에 어긋나기에 오답으로 처리하였으며, 각 동형이의어번호는 확실히 잘못 태깅되었다. 예에서 '공표하'는 어간형인 '공표하/VV'를 정답으로 인정하며, 만약 어근형을 기준으로 하더라도 '공표_01/NNG+하/XSV'가 정답이다.

5. 결 론

형태소 분석 및 의미결정에는 후보를 선택하는 태깅 과정이 있으며, 본 논문은 태깅에서 부분어절 조건부확률 모델(SCP)을 사용하는 알고리즘을 제안한다. 기존에는 은닉 마르코프 모델(HMM)이 주로 사용되었으나 어절당 후보 수의 제곱에 비례하는 연산량 때문에 속도가 느리다는 단점이 있다. SCP는 정확률이 비록 HMM보다 낮지만 그 차이는 0.07%로 매우 작으며, 속도는 2배 이상 빠른 장점을 보였다.

SCP는 어절의 처음 또는 끝 2음절을 이용하는 경우가 많기 때문에 길이가 3음절 이상인 형태소에서 정확률이 낮아지는 경향을 보였다. 이것은 후보에서 형태소의 길이가 길수록 가중치를 올리는 방식으로 문제를 완화시킬 수 있을 것이다. 이에 관한 자세한 연구가 필요할 것으로 판단된다. HMM과 마찬가지로 SCP도 학습용량이 지나치게 커지는 경향을 보였다. 이를 해결하기 위해서 빈도가 낮은 정보나 일부 중복성이 강한 정보를 삭제하는 방식을 도입해볼 필요가 있다.

현재 자연어처리에서 많은 연구가 되고 있는 Conditional Random Field(CRF)와 본 연구를 비교해보고 서로의 장점을 조합한 새로운 알고리즘을 연구해볼 수도 있다. SCP에서 사용한 자질들을 CRF에서 사용해볼 수 있을 것이며, CRF에서 사용하는 가중치 결정 방식(최적화 알고리즘)을 SCP에 응용할 수도 있을 것이다.

한국어 어휘의미망(U-WIN)은 동형이의어 구분에 사용될 수 있는 매우 유용한 정보들을 가지고 있으나 본 논문은 말뭉치와 표준국어대사전 정보만을 다루고 있다. 만약 U-WIN

을 같이 사용한다면 동형이의어 분석 정확률이 향상될 것이다. 이를 위해 어휘의미망을 활용한 기존의 연구들을 참조할 수 있으며, 이런 연구들과 형태소 분석기의 조합 방법은 충분히 연구할 가치가 있다.

References

[1] Jin-dong Kim, Heui-Seok Lim, and Hae-Chang Rim, "Twoply HMM: A Part-of-Speech Tagging Model based on Morpheme-Unit considering the Characteristics of Korean", *Journal of KIISE*, Vol.24, No.12, pp.1502-1512, Dec., 1997.

[2] Hee-Geun Park, Y. M. Ahn, and Y. H. Seo, "Korean Part-of-Speech Tagging System Using Resolution Rules for Individual Ambiguous Word(in Korean)", *Journal of KIISE: Computing Practices and Letters*, Vol.13, No.6, pp.427-431, 2007.

[3] Scott M. Thede, Mary P. Harper, "A Second-Order Hidden Markov Model for Part-of-Speech Tagging", In *Proceedings of the 37th of ACL*, pp.175-182, 1999.

[4] Eric Brill, "Transformation-based error-driven learning and natural language processing: A case study in part of speech tagging", *Computational Linguistics*, Vol.21, No.4, pp.543-565. 1995.

[5] Dan Roth, Dmitry Zelenko, "Part of speech tagging using a network of linear separators", Proceedings of COLING-ACL 98, pp.1136-1142, 1998.

[6] David Yarowsky, "Word-sense disambiguation using statistical models of Roget's categories trained on large corpora", International Conference On Computational Linguistics(Proceedings of the 14th conference on Computational linguistics-Vol.2), pp.454-460, 1992.

[7] Soojong Lim, Youngja Park, and Mansuk Song, "Word Sense Disambiguation of Korean Verbs Using Weight Information from Context", In Proceedings of the 10th Conference on Hangul and Korean Language Information Processing, pp.425-429, Oct., 1998.

[8] Jun-Su Kim, H. S. Choe, and C. Y. Ock, "A Korean Homonym Disambiguation Model Based on Statistics Using Weights(in Korean)", *Journal of KIISE: Software and Applications*, Vol.30, No.11, 2003.

[9] Wang Woo Lee, "Word Sense Disambiguation System Using Lexical Co-occurrencing Set and Thesaurus(in Korean)", Master Thesis, Ulsan university, 2003.

[10] Yong-Gu Lee, Y. M. Chung, "An Experimental Study on an Effective Word Sense Disambiguation Model Based on Automatic Sense Tagging Using Dictionary Information (in Korean)", *Journal of the Korean Society for Information Management*, Vol.24, No.1, 2005.

[11] Jeong Heo, H. C. Seo, and M. G. Jang, "Homonym Disambiguation based on Mutual Information and Sense-Tagged Compound Noun Dictionary(in Korean)", *Journal of KIISE: Software and Applications*, Vol.33, No.12, 2003.

[12] Dong Myung Kim, "Simultaneous Korean POS and Homonym Tagging System using HMM(in Korean)", Masters Thesis, Ulsan University, 2009.

[13] Minho Kim, H. C. Kwon, "Word Sense Disambiguation using Semantic Relations in Korean WordNet(in Korean)", *Journal of KIISE: Software and Applications*, Vol.38, No.10, pp.503-577, 2011.

[14] Young-Jun Base, Cheol-Young Ock, "Semantic Analysis of Korean Compound Noun using Lexical Semantic Network(U-WIN)", Ph. D. Thesis, Ulsan University, 2013.

[15] Joon-Choul Shin, Cheol-Young Ock, "A Stage Transition Model for Korean Part-of-Speech and Homograph Tagging", *Journal of KIISE: Software and Applications*, Vol.39, No.11, pp.889-901, 2012.

[16] Joon-Choul Shin, Cheol-Young Ock, "Comparison between Markov Model and Hidden Markov Model for Korean Part-of-Speech and Homograph Tagging", In Proceddings of the 25th Conference of Hangul and Korean Information Processing, pp.152-155, Oct., 2013.

[17] Joon-Choul Shin, C. Y. Ock, "A Korean Morphological Analyzer using a Pre-analyzed Partial Word-phrase Dictionary(in Korean)", *Journal of KIISE*, Vol.39, No.5, 2012.

[18] Ho Suk Lee, "A Survey of conditional Random Fields and Applications", In Proceddings of Fall Conference of KIISE, Vol.36, No.2, 2009.

[19] Seung-Hoon Na, Chang-Hyun Kim, and Young-Kil Kim, "Semi-CRF or Linear-chain CRF? A comparative Study of Joint Models for Korean Morphological Analysis and POS Tagging", In Proceddings of the 25th Conference of Hangul and Korean Information Processing, pp.9-12, 2013.

신 준 철

e-mail : ducksjc@nate.com

2007년 울산대학교 컴퓨터정보통신공학과 (학사)

2009년 울산대학교 컴퓨터정보통신공학과 (석사)

2014년 울산대학교 컴퓨터정보통신공학과 (박사)

2011년~현 재 울산대학교 지능형컴퓨터연구실 연구교수

관심분야 : Korean Language Processing, Document Clustering

옥 철 영

e-mail : okcy@ulsan.ac.kr

1982년 서울대학교 컴퓨터공학과(학사)

1984년 서울대학교 컴퓨터공학과(석사)

1993년 서울대학교 컴퓨터공학과(박사)

1994년 러시아 TOMSK 공과대학 교환교수

1996년 영국 GLASGOW 대학교 객원교수

2007년~2008년 한국정보과학회 언어공학연구회 위원장

2008년 국립국어원 객원교수

1984년~현 재 울산대학교 전기공학부 컴퓨터정보통신공학전공 교수

2010년~현 재 울산대학교 국어국문학부 겸직교수

관심분야 : Korean Language Processing, Korean Homograph Tagging, Ontology, Knowledge Base, Document Clustering