

CRFs와 TBL을 이용한 자동화된 음성인식 후처리 방법

(An Automatic Post-processing Method for Speech Recognition using CRFs and TBL)

선충녕[†] 정형일[†] 서정연^{**}
(Choongnyoung Seon) (Hyoungil Jeong) (Jungyun Seo)

요약 음성 인식기의 오류는 음성기반 응용 시스템들의 성능에 크게 영향을 주기 때문에 오류를 줄이기 위한 효과적인 처리 방법이 필요하다. 기존의 후처리 기법들은 수동 작업을 통한 코퍼스나 규칙으로 후처리를 수행하는 것이 일반적이다. 본 논문에서는 문제나 인식기의 특성에 무관하게 자동으로 학습할 수 있는 후처리 모델을 제안한다. 후처리의 문제를 오류의 인식과 수정으로 구분하고 오류 검출 문제는 순차적인 분류 문제로 간주하여 conditional random fields(CRFs)를 사용하고 오류 수정 규칙은 transformation-based learning(TBL)을 이용하여 자동 생성하여 적용하였다. 제안한 방법을 여행 예약 영역의 음성 인식기에 적용한 결과 삽입, 삭제, 치환 오류를 각각 25.85%, 3.57%, 7.42%을 수정하였으며, 이로 인해 인식기의 어휘 오류율을 2% 감소시킬 수 있었다.

키워드 : 음성 인식, 음성 인식 후처리, 조건적 임의 필드, 변형 기반 학습

Abstract In the applications of a human speech interface, reducing the error rate in recognition is the one of the main research issues. Many previous studies attempted to correct errors using post-processing, which is dependent on a manually constructed corpus and correction patterns. We propose an automatically learnable post-processing method that is independent of the characteristics of both the domain and the speech recognizer. We divide the entire post-processing task into two steps: error detection and error correction. We consider the error detection step as a classification problem for which we apply the conditional random fields (CRFs) classifier. Furthermore, we apply transformation-based learning (TBL) to the error correction step. Our experimental results indicate that the proposed method corrects a speech recognizer's insertion, deletion, and substitution errors by 25.85%, 3.57%, and 7.42%, respectively.

Key words : Speech Recognition, Post-processing for Speech Recognition, Conditional Random Fields, Transformation-Based Learning

· 이 논문은 한국연구재단의 중견연구자 프로그램의(No. 2009-0086194) 지원으로 이루어진 결과의 일부입니다.

† 학생회원 : 서강대학교 컴퓨터공학과
wilowisp@sogang.ac.kr
hijeong@sogang.ac.kr

** 종신회원 : 서강대학교 컴퓨터공학과 교수
seojy@sogang.ac.kr
논문접수 : 2010년 1월 19일
심사완료 : 2010년 7월 6일

Copyright©2010 한국정보과학회: 개인 목적이나 교육 목적인 경우, 이 저작물의 전체 또는 일부에 대한 복사본 혹은 디지털 사본의 제작을 허가합니다. 이 때, 사본은 상업적 수단으로 사용할 수 없으며 첫 페이지에 본 문구와 출처를 반드시 명시해야 합니다. 이외의 목적으로 복제, 배포, 출판, 전송 등 모든 유형의 사용행위를 하는 경우에 대하여는 사전에 허가를 얻고 비용을 지불해야 합니다.

정보과학회논문지: 소프트웨어 및 응용 제37권 제9호(2010.9)

1. 서론

최근의 음성 인식 기술의 향상에도 불구하고, 여전히 음성 인식 결과는 많은 오류를 포함하고 있다. 통계기반 음성 인식 결과의 오류를 감소시키기 위해서는, 대용량의 학습 데이터를 추가로 구축하는 작업이 필요하다. 그러나, 이러한 작업은 고비용의 인적, 물적 자원을 요구한다. 이러한 문제점을 극복하기 위해 다양한 후처리 연구들이 진행되고 있다. Ringger와 Allen은 입력 문장이 오류를 포함한 문장으로 인식되는 것을 잡음 채널 모델(Noisy Channel Model)로 해석하였다[1]. 그러나 이 시스템은 치환 에러를 제외한 삭제, 삽입 에러에 취약한 문제가 있었다. Kaki et al.는 오류 패턴 매칭 기법을

이용한 후처리 시스템을 제안하였다[2]. 수동으로 구축된 규칙에 의한 후처리는 도메인이 작을 때 유용하고 직관적이기 때문에 관리가 용이하지만, 인식기나 환경에 의존적이기 때문에 영역 확장성이 부족했다. 김용현과 정민화는 오류 패턴의 확률을 계산하고, 문맥 정보를 이용하는 통계적인 방법의 오류 보정 시스템을 제안하였다[3]. 이 시스템은 무조건 적용되는 규칙 기반 방법의 문제점을 확률을 이용하여 위험성을 줄였으나 인식기의 결과가 N-best 형태로 제공될 때만 적용할 수 있는 단점이 있었다.

후처리를 위한 최근 연구들은 종래의 인식 결과 단어 열 외에 추가적인 정보를 이용하여 성능을 높이는 방향으로 진행되고 있다. Jeong et al.은 어휘 의미 패턴(Lexico-Semantic Pattern)을 사용한 오류 수정 시스템을 제안하였다[4,5]. 이 모델들은 어휘들이 가지는 의미적 범주를 사용하여 오류의 수정 성능을 높였다. 그러나 이 접근 방법은 특정한 의미 범주를 가지지 못하는 단어의 오류를 처리하기 힘들고, 영역 변경 시에 관련된 의미 범주화 작업이 필요하여 확장성에 한계가 있다. Lopez-Cozar와 Callejas는 대화 영역에서 문맥과 상황에 맞춘 후처리를 제안하였다[6]. 이 방법은 인식 어휘의 신뢰도를 정보로 사용하고 어휘, 문법, 의미, 문맥적인 정보를 이용하여 후 처리의 성능을 향상시켰다. 하지만 이 접근 방법은 다양한 형태의 정보를 사용함으로써 도메인 변화 시 코퍼스 구축에 노력이 필요하며, 문맥정보라는 특별한 형태의 정보를 사용함으로써 응용 영역이 대화에 제한되는 한계가 있다.

최근에는 높은 후처리 성능을 위해 영역 특성을 고려한 후처리 기법들이 연구되고 있지만, 이런 기법들은 그만큼 많은 양의 수동 작업이 요구된다. 수동 작업의 부담은 영역이나 인식기 변화에 따라 비용을 요구하기 때문에 활용도가 낮아진다. 따라서 우리는 기계 학습 방법을 이용하여 수동 지식이나 외부 지식 없이 인식기의 성능을 높여줄 수 있는 후처리 방법을 제안한다. 제안 방법은 음성 인식기 학습을 위해 이용된 음성 전사 데이터와 인식기의 출력만을 사용하고 추가적인 수동 작업을 요구하지 않는다. 또한, 영역이나 인식기의 특성을 사용하지 않기 때문에 다른 영역에도 재 학습을 통해 바로 이식할 수 있다는 장점이 있다.

본 논문의 구성은 다음과 같다. 2장에서 본 논문에서 제안한 음성인식 후보정 시스템에 대해 설명한다. 3장에서는 연속음성 인식기의 결과에 제안한 후처리 시스템을 적용하여 시스템의 오류 보정 능력을 실험 및 평가하고, 4장에서는 결론 및 향후 과제를 보인다.

2. 음성인식 후처리

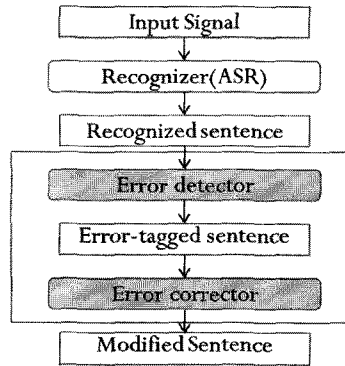


그림 1 음성 인식 후처리

제안하는 방법의 전체 구조는 그림 1과 같다.

2.1 확장 편집 연산

우리는 음성 인식 결과 단어 열에서 오류를 검출하기 위해 음성의 전사 데이터와 인식된 결과를 사용한다. 학습 시점에서의 오류는 전사 문장에 포함된 단어 열과 다르게 인식된 결과 열이라고 정의할 수 있다. 주어진 두 입력 열의 차이를 비교하고 오류를 찾기 위해 우리는 편집 연산(Edit operation)을 이용한다. 편집 연산은 하나의 문자열이 다른 문자열과 같아지기 위해 필요한 연산이다. 편집 연산은 하나의 단어가 다른 단어로 치환된 경우(SUB), 한 쪽의 단어가 삭제된 경우(DEL), 그리고 불필요한 단어가 삽입된 경우(INS)로 구성된다. 하지만 편집 연산을 오류 판별을 위해 사용하는 경우 한 가지 문제점이 있다. 그 문제점은 아래의 그림 2와 같이 음성 인식 열에서 특정 단어가 삭제된 경우, 편집 연산을 적용하면 삭제(DEL) 연산을 부여할 어휘가 없으므로 삭제 오류에 대해 처리할 수 없다는 점이다.

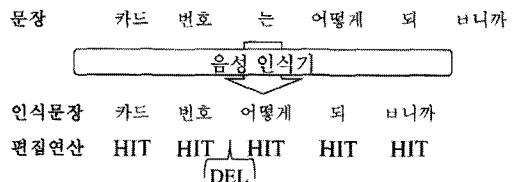


그림 2 편집 연산 적용시 문제점

이 문제를 위해 편집 연산을 삭제 위치를 판단할 수 있도록 변형하였다. 편집 연산에서 삭제(DEL)를 제외한 3가지 연산(HIT, SUB, INS)을 좌우에 삭제와 함께 나타낼 수 있도록 확장하였다. 아래의 표 1은 확장된 편집 연산을 보여주고 있다.

학습을 위해 예문을 확장된 편집 연산 범주를 적용한 결과는 다음과 같다.

표 1 삭제제를 제외한 확장 편집 연산 범주

편집연산	설명
HIT	현재 단어 일치
_HD	현재 단어 일치, 다음 단어 삭제
DH_	현재 단어 일치, 이전 단어 삭제
DHD	현재 단어 일치, 좌우 단어 삭제
SUB	현재 단어 치환
_SD	현재 단어 치환, 다음 단어 삭제
DS_	현재 단어 치환, 이전 단어 삭제
DSD	현재 단어 치환, 좌우 단어 삭제
INS	현재 단어 삽입
_ID	현재 단어 삽입, 다음 단어 삭제
DI_	현재 단어 삽입, 이전 단어 삭제
DID	현재 단어 삽입, 좌우 단어 삭제

편집연산	HIT	_HD	DH_	HIT	HIT
인식열	카드	번호	어떻게	되	비니까

2.2 오류 검출기(Error Detector)

오류 검출기는 인식 결과 어휘열이 주어졌을 때 그 어휘 중에서 오류 어휘를 찾는 역할을 한다. 이를 위해 음성 인식기의 오류를 찾는 문제를 연속된 어휘 열에서 오류 어휘를 추출하는 문제로 대응하였다. 입력된 문장 s는 단어들의 열 $W = \{w_1, \dots, w_N\}$ 이라고 할 때 오류 검출기는 오류를 판별할 수 있는 확장 편집 연산의 열인 $T = \{t_1, \dots, t_N\}$ 를 할당한다. 학습 단계에서 코퍼스로 사용될 편집 연산이 부착된 인식 단어열은 인식 단어열과 전사 단어열을 이용하여 자동으로 구축하였다.

각각의 단어에 대해 범주를 결정하는 문제에서 오류 열은 주변 문맥에서 오류의 발생 여부와 그 종류에 영향을 받는다. 특히 삭제 위치를 명확하게 찾기 위해서는 연속적인 편집 연산들이 접속 가능하게 나타나는 것이 중요하다. 따라서 우리는 기계 학습 방법 중에서 연속적인 범주 할당 문제에서 좋은 성능을 보여주고 있는 Conditional Random Fields(CRFs)[7] 모델을 이용하였다. CRFs는 기존의 연속적인 범주 할당 문제에서 사용되던 은닉 마르코프 모델이 가지고 있던 독립 가정을 완화시키고 이전 범주에 의해 범주 결정이 크게 영향을 받는 레이블 편향 문제를 극복한 모델이다. 우리는 주어진 인식 단어 열 W 에 대해 CRFs를 이용하여 식 (1) 과 같이 조건부 확률 값을 계산한다.

$$P(T|W) = \frac{1}{Z(X)} \exp\left(\sum_{i=1}^N \sum_k \lambda_k f_k(t_{i-1}, t_i, w, i)\right) \quad (1)$$

식 (1)에서 t_i 는 i 번째 편집 연산을 의미하고 w 는 인식 어휘를 나타낸다. $f_k(t_{i-1}, t_i, x, i)$ 는 자질 함수를 의미하며, λ_k 는 자질의 가중치이다. 식 (1)에서 볼 수 있듯이 CRFs는 t_{i-1} 에서 t_i 로의 전이 자질을 사용하여 코퍼

표 2 오류 검출을 위한 자질 집합

Unigram 자질	Bigram 자질
w_{i-2}	$w_{i-2} w_{i-1}$
w_{i-1}	$w_{i-1} w_i$
w_i	$w_i w_{i+1}$
w_{i+1}	$w_{i+1} w_{i+2}$
w_{i+2}	

스에서 출현한 편집 연산의 조합으로 결과를 제공할 수 있다. 자질은 슬라이딩 윈도우와 어휘 n-gram을 이용하여 구축하였다. 윈도우의 크기는 현재 위치를 기준으로 5를 사용하고 그 안에서 unigram과 bigram 자질을 이용하였다. 위치 i 에서 이용된 자질의 집합은 다음의 표 2와 같다. 표 2에서 w_i 는 인식 어휘 열에서 i 번째에 위치한 어휘를 의미한다.

음성 인식 결과열은 오류 검출기를 통해 확장된 편집 연산의 열이 제공된다. 이 결과에 따라 삭제 위치를 파악할 수 있으므로, 일반적인 편집 연산 열로 변환되어 오류 수정기에 제공된다. 이때 삭제 연산에 대응되는 단어로는 ‘_’을 사용하였다.

2.3 오류 수정기(Error Corrector)

오류 검출기에 의해 음성 인식 문장에서 발견된 오류는 위치와 종류만 결정되기 때문에 발견된 오류를 수정하기 위한 방법이 필요하다. 오류 수정 규칙을 수동으로 구축하는 것이 정확하고 직관적이지만 수동 구축된 규칙들은 인식기나 영역에 의존적이기 때문에 확장성이 부족하다. 따라서 이 문제를 해소하기 위해 규칙을 자동으로 학습하는 방법을 사용하였다. 코퍼스에서 자동으로 오류 수정 규칙을 학습하기 위해 우리는 변형 기반 학습(TBL: Transformation-based Learning)[8]을 이용하였다. TBL은 정답에 가까워 질 때까지 오류를 줄인다는 아이디어를 이용한 규칙 학습 방법이다. TBL은 입력에 초기 정보 부착을 수행하고 매 단계마다 오류 수정 빈도와 잘못된 수정의 빈도의 차이가 가장 적은 규칙을 하나씩 선정하여 변형 규칙 리스트에 순차적으로 저장된다. TBL은 품사 태깅, 개체명 인식 등 다양한 분야에 이용되었는데 주로 입력 어휘에 대해 범주를 할당하는 용도로 이용되었다. 그러나 본 논문의 경우 어휘를 직접 수정하기 위한 용도로 사용되기 때문에 오류 검출기에서 제공되는 편집 연산 열을 입력으로, 인식 열을 수정하여 정답에 맞게 변형한다. TBL은 규칙을 생성할 수 있는 템플릿을 작성하여 자동으로 규칙을 학습한다. 우리는 오류 발생 위치의 좌우 어휘를 고려하여 아래의 표 3과 같은 총 65개의 템플릿을 작성하여 TBL 규칙을 학습하였다. 표 3에서 w 는 어휘를 의미하며, e 는 편집 연산을 의미한다. 첨자는 현재 위치를 기준으로

표 3 TBL 규칙 템플릿

편집연산	어휘	템플릿
e_{-2} e_{-1} e_0 e_1 e_2	w_0 $w_{-1} w_0$ $w_0 w_1$ $w_{-2} w_{-1}$ $w_1 w_2$	$e_{-2} w_0$ $e_{-2} w_{-1} w_0$ $e_{-2} w_0 w_1$ $e_{-2} w_{-2} w_{-1}$ $e_{-2} w_1 w_2$ $e_{-1} w_0$ $e_{-1} w_{-1} w_0$ $e_{-1} w_2 w_{-1}$ $e_0 w_0$ $e_0 w_1 w_0$ $e_0 w_0 w_1$ $e_0 w_{-1} w_0$ $e_0 w_1 w_2$ $e_0 w_2 w_{-1}$ $e_1 w_0$ $e_1 w_{-1} w_0$ $e_1 w_1 w_2$ $e_2 w_0$ $e_2 w_1 w_0$ $e_2 w_0 w_1$ $e_2 w_{-1} w_0$ $e_2 w_1 w_2$
$e_{-1} e_0$ $e_0 e_1$ $e_{-1} e_1$ $e_{-1} e_{-1}$ $e_1 e_2$	w_0 $w_{-1} w_0$ $w_0 w_1$ $w_{-2} w_{-1}$ $w_1 w_2$	$e_{-1} e_0 w_0$ $e_{-1} e_0 w_{-1} w_0$ $e_{-1} e_0 w_0 w_1$ $e_{-1} e_0 w_{-2} w_{-1}$ $e_{-1} e_0 w_1 w_2$ $e_0 e_1 w_0$ $e_0 e_1 w_{-1} w_0$ $e_0 e_1 w_0 w_1$ $e_0 e_1 w_2 w_{-1}$ $e_0 e_1 w_1 w_2$ $e_0 e_1 w_0 w_1$ $e_0 e_1 w_{-1} w_0$ $e_0 e_1 w_0 w_1$ $e_0 e_1 w_2 w_{-1}$ $e_0 e_1 w_1 w_2$ $e_1 e_2 w_0$ $e_1 e_2 w_1 w_0$ $e_1 e_2 w_0 w_1$ $e_1 e_2 w_{-1} w_0$ $e_1 e_2 w_1 w_2$
$e_{1,3,-1}$ $e_{1,3}$	w_0 $w_{-1} w_0$ $w_0 w_1$ $w_{-2} w_{-1}$ $w_1 w_2$	$e_{1,3,-1} w_0$ $e_{1,3,-1} w_{-1} w_0$ $e_{1,3,-1} w_0 w_1$ $e_{1,3,-1} w_{-2} w_{-1}$ $e_{1,3,-1} w_1 w_2$ $e_{1,3} w_0$ $e_{1,3} w_{-1} w_0$ $e_{1,3} w_0 w_1$ $e_{1,3} w_{-2} w_{-1}$ $e_{1,3} w_1 w_2$
\emptyset	$w_{-1} w_0$ $w_0 w_1$ $w_{-2} w_{-1} w_0$ $w_{-1} w_0 w_1$ $w_0 w_1 w_2$	$w_{-1} w_0$ $w_0 w_1$ $w_{-2} w_{-1} w_0$ $w_{-1} w_0 w_1$ $w_0 w_1 w_2$

0은 현재 위치, -1은 이전위치, 1은 다음위치를 의미한다. [1,3]과 같은 표현은 범위 표현으로 $e_{1,3}$ 은 현재 위치로부터 다음 3개의 편집 연산들 중에서 특정 연산이 출현했다는 것을 의미한다.

그림 3은 TBL에 의해 자동 학습된 규칙 열에 의해 오류 어휘 열이 맞게 고쳐지는 과정을 보여준다.

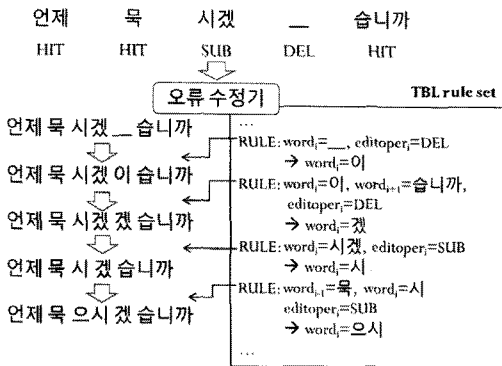


그림 3 TBL 기반 오류 수정기 동작 예

학습 과정에서 순서에 따라 구축된 TBL 규칙 집합은 순차적으로 입력에 적용된다. 그림 3의 TBL 규칙에서 word는 인식 단어, editoper는 편집 연산 그리고 첨자는 발생한 위치를 의미한다. 그림 3에서 적용된 첫 번째 규칙은 “현재 어휘가 ‘_’이고 편집 연산이 삭제이면 어휘를 ‘이’로 치환하라”는 것을 의미한다. 그림 3에서는 현재 입력 문자열에 적용될 수 있는 규칙들을 순차적으로 적용시키면 치환과 삭제 오류를 포함하고 있는 문장이 정답 문장으로 변환되는 과정을 볼 수 있다.

3. 실험결과

3.1 실험 환경

제안하는 시스템의 오류 검출 및 보정의 성능 평가를 위해 이경남의 연속 음성 인식기[9]에 후처리를 적용하였다. 이경남의 연속음성 인식기는 형태소 단위 연속 음성을 인식하고 종래의 인식기에 비해 높은 성능을 보여준다. 인식기에 적용한 데이터는 호텔 예약, 항공 예약, 여행 정보 분야에서 사람들 사이의 전화 대화를 전사한 것으로 총 1,054 문장(14,851어휘)을 수집하여 실험하였다. 실험은 5차 교차 비교 검증을 통해 학습과 실험 데이터로 나누어 진행하였으며 오류 수정 성능을 평가하기 위해 어휘 오류율(WER: Word Error Rate)을 사용하였다. 어휘 오류율은 전사 데이터에 나타난 정답 어휘수에 대한 인식 문장에 포함된 오류(삽입, 삭제, 치환)의 비율을 나타내는 값으로 식 (2)와 같이 계산된다.

$$WER = \frac{(SUB + INS + DEL)}{\text{정답어휘수}} \times 100 \quad (2)$$

3.2 실험 결과

음성 인식기에 제안한 후처리 방법을 적용하기 전과 후의 어휘 오류율은 아래 표 4와 같다.

표 4 제안 후처리 적용에 따른 어휘 오류율

시스템	정답 어휘 수	전체오류 어휘 수	WER (%)
Lee,2006	2688.4	410.4	15.23
Lee,2005 + 후처리	2688.4	355.8	13.23

제안한 후처리 시스템은 전체적인 WER을 2.00% 낮추는 성능을 보였다. 제안 방법을 사용하였을 때 오류별 수정률은 아래의 표 5와 같다. 표 5에서 제안 모델은 3가지 오류 유형을 모두 수정하고 있으며, 특히 삽입 오류에 대해 효과적으로 처리함을 보여준다.

표 5 오류 유형별 평균 수정률

오류 유형	수정 전 오류빈도(f)	수정 후 오류빈도(c)	수정률 (%) (((f-c)*100)/f)
SUB	212.6	196.2	7.42
INS	134.4	99.2	25.85
DEL	63.4	60.4	3.57

3.3 오류 분석

제안한 후처리 시스템은 코퍼스에서 자주 발생하는 오류에 대해 좋은 결과를 보여준다. 특히 아래의 예에서 보여주는 것처럼 축약어나 어미의 오인식 등 기능어에서의 다양한 오류를 효과적으로 처리하고 있다.

정답 동학사 는 어떠하 **ㄴ** **ㅈ** **이** **비** **나** **까**
 인식 동학사 는 어떤 **것** **습** **니** **까**
 인식태그 HIT HIT HIT SUB _SD
 편집태그 HIT HIT HIT SUB SUB DEL
 후처리 동학사 는 어떤 **ㅈ** **이** **비** **나** **까**

하지만 주변 어휘 정보를 이용하여 자동으로 학습된 규칙을 사용하기 때문에 아래의 예처럼 연속된 오류가 나타난 경우에 문제점이 발견되었다.

정답 작석 을 몇 개 **예** **약** **하** **시** **ㄹ** **예** **정** **이** **시** **비** **나** **까**
 인식 작석 을 몇 **기** **르** **가** **시** **ㄹ** **예** **정** **이** **있** **습** **니** **까**
 인식태그 HIT HIT HIT HIT HIT HIT HIT HIT HIT HIT HIT SUB SUB
 편집태그 HIT HIT HIT HIT HIT HIT HIT HIT HIT HIT SUB SUB
 후처리 작석 을 몇 **기** **르** **가** **시** **ㄹ** **예** **정** **이** **시** **비** **나** **까**

이 문제는 학습 데이터 부족으로 인해 발생했다. 실제로 실험에 사용한 학습 문장의 수는 평균 843 문장으로 매우 적었기 때문에 학습된 수정 규칙은 평균 205.8개로 제한적이었다. 제안 방법은 코퍼스 구축이 쉽기 때문에 학습 데이터를 충분히 확보함으로써 문제를 최소화될 수 있다. 또한 어휘보다 작은 단위인 음소, 발음열 등을 자질로 이용하는 방법에 대해 연구한다면 더 효과적인 해결이 가능할 것이다.

3.4 후처리 성능 비교

우리는 제안 모델의 효율성을 보이기 위해 기존 연구들의 결과와 비교하였다. 아래 표 6은 기존 오류에 대한 수정된 오류의 비율인 오류 수정률을 기준으로 기존 방법과 제안한 방법의 성능을 비교하고 있다.

표 6 기존 방법론과의 비교

자질 특성	후처리 방법	오류 수정률	인식기(오류율)
어휘 정보	오류 패턴 [2]	8.5%	Masataki (25.27%)
	통계적 예러 패턴 [3]	12.1%	Word bigram model
	제안 방법	13.13%	이경님[9] (15.23%)
어휘 + 확장 정보	의미기반 오류 수정 [4]	36.70% 30.29%	LG-Elite(20.49%) ByVoice(18.75%)
	대화 발화 오류 수정 [6]	35.59%	HTK-based (23.88%)

후처리 시스템의 경우 기반 인식기의 성능과 영역의 특성에 영향을 받기 때문에 서로 다른 인식기와 영역에 적용된 후처리 방법을 객관적으로 비교하는 것은 힘들다. 그러나 오류 수정률의 입장에서, 제안 방법은 기반 인식기 성능이 높고, 수동 작업을 최소화하였음에도 어휘 수준의 정보만을 사용하는 다른 방법론들과 비슷하거나 높은 효율성을 보여주고 있음을 볼 수 있다. 또한 높은 이식성을 가지고 있기 때문에 확장 정보를 사용한 모델들과도 비교할 만한 성능이라고 판단된다.

4. 결론 및 향후 과제

본 논문은 CRFs와 TBL을 이용하여 음성 인식기의 후처리 방법을 제안하였다. 제안 방법은 전사된 문장 정보와 인식 결과 만을 가지고 모든 과정이 자동으로 진행된다. 따라서 코퍼스 개발에 수동 노력이 필요하지 않고, 영역 전이나 인식기 변화에도 재 학습 과정을 통해 쉽게 적용 될 수 있다. 실험에서는 자료 부족에 의한 문제점들이 일부 보였다. 이 단점은 쉬운 코퍼스 구축 방법으로 일부 완화될 수 있다. 또한 향후 음절 및 발음 수준의 수정 규칙 학습 방법에 대한 연구가 추가적으로 진행된다면 보다 높은 성능을 기대할 수 있을 것이다.

참고 문헌

- [1] E. K. Ringger and J. F. Allen, "A Fertility Channel Model for Post-Correction of Continuous Speech Recognition," *Proceedings of the Fourth International Conference on Spoken Language Processing (ICSLP96)*, vol.2, pp.897-900, 1996.
- [2] S. Kaki et al., "A Method for Correcting Errors in Speech Recognition Using the Statistical Features of Character Co-occurrence," *Proceedings of the 17th international conference on Computational linguistics*, vol.1, pp.653-657, 1998.
- [3] Y. Kim, M. Jeong, "Improving Performance of Continuous Speech Recognition Using Error Pattern Training and Post Processing Module," *Proceedings of the KIISE Korea Computer Congress 2000*, vol.27, no.1, pp.441-358, 2000. (in Korean)
- [4] M. Jeong, B. Kim, G. G. Lee, "Semantic-Oriented Error Correction for Spoken Query Processing," *Proceedings on IEEE Automatic speech recognition and understanding workshop (ASRU2003)*, pp.156-161, 2003.
- [5] M. Jeong, S. Jung, G. G. Lee, "Speech recognition error correction using maximum entropy language model," *Proceedings of Interspeech*, pp.2137-2140, 2004.
- [6] R. López-Cózar, Z. Callejas, "ASR post-correction for spoken dialogue systems based on semantic, syntactic, lexical and contextual information," *Speech Communication*, vol.50, Issue.8-9, pp.745-766, 2008.
- [7] J. Lafferty et al., "Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data," *Proceedings of ICML*, pp.282-289, 2001.
- [8] E. Brill, "A Simple Rule-based Part of Speech Tagger," *Proceedings of the Third Conference on Applied Natural Language Processing*, pp.152-155, 1992.
- [9] K. Lee, *Morph-Phonological Modeling of Pronunciation Variation for Korean Large Vocabulary Continuous Speech Recognition*, Ph.D Thesis, Sogang University, 2006.



선 충 념

1999년 서강대학교 컴퓨터학과 학사. 2001년 서강대학교 대학원 컴퓨터학과 석사. 2002년~2006년 (주) 다이퀘스트 연구소 선임연구원. 2001년~현재 서강대학교 대학원 컴퓨터공학과 박사과정. 관심분야는 대화 시스템, 정보 검색, 개체명 인식,

형태소 분석



정 형 일

2006년 서강대학교 컴퓨터학과 학사. 2008년 서강대학교 대학원 컴퓨터학과 석사. 2008년~현재 서강대학교 대학원 컴퓨터공학과 박사과정. 관심분야는 정보 요약, 정보 추출, 정보 검색



서 정 연

1981년 서강대학교 수학과 학사. 1985년 미국 Univ. of Texas, Austin 전산학과 석사. 1990년 미국 Univ. of Texas, Austin 전산학과 박사. 1990년~1991년 미국 Texas Austin, UniSQL Inc. Senior Researcher. 1991년 한국과학기술원 인

공지능 연구센터 선임연구원. 1991년~1995년 한국과학기술원 전산학과 교수. 1995년~현재 서강대학교 컴퓨터공학과 교수. 관심분야는 한국어 정보처리, 자연어처리, 기계번역, 대화처리