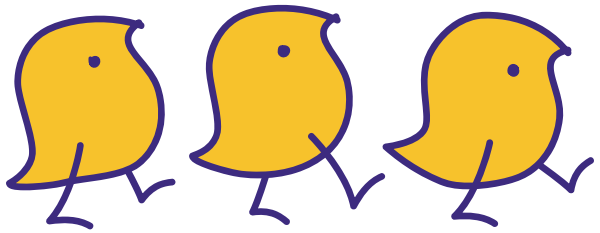
LangCon 2019

스마트 스피커에서의 음악 재생 발화 오류 교정

카카오 고병일

목차



스마트 스피커란?
음성 가상 비서

LangCon 2019

그래도 스마트 스피커가 뭔지는 알아보고 가자

Language Conference 2019

스마트 스피커

- 가상 비서 서비스 제품
- 인공지능 기술 적용
- **음성 명령**을 통해 음악 감상, 정보 검색 등의 비서기능을 수행하는 기기
  - 카카오 미니
  - NAVER 클로바
  - AMAZON Echo(알렉사)
  - Google 구글홈(Google Assistant)

왜 스마트 스피커 일까?(WHY)

Why Smart Speaker?
음성을 이용한 가상 비서 서비스의 첫 진입단계
거실에서 누구나 쉽게 접근 가능

Next?
가상비서플랫폼을 장악
➔ IoTs를 장악가능

Then?
여기엔 다른 서비스가 들어 올수 없음

https://www.huffingtonpost.com/samcohen/internet-of-things-as-the_b_10937956.html

왜 스마트 스피커 일까?(cont`)

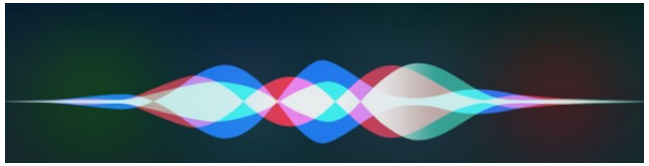
아마존 에코를 사용하면..

- “알렉사 식료품 주문해줘” ➔ 아마존 쇼핑
- “알렉사 마이클 잭슨 노래 들려줘” ➔ 아마존 뮤직
- “알렉사 어벤져스 틀어줘” ➔ 아마존 프라임 비디오
- “알렉사 피자헛에서 피자주문해줘” ➔ 피자헛 (3rd party skills)
  - 자동차, 가전, 회사들 참여..

➔ 2등은 없음(?) / 모든 것을 승자 독식이 가능

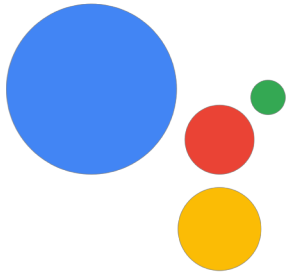
History

IBM Shoebox
1962년
첫 디지털 음성 인식 도구



Apple Siri
2010년
현대적인 첫 가상 비서 서비스

Amazon Echo
- Alexa, 2014년 10월



Google Assistant(Home)
- Hey Google, 2016년 5월

NUGU

SKT NUGU
Aria, 2016년 9월

Clova

Naver Clova
- Clova, 2017년 5월

Kakako Mini
- Hey Kakako, 2017년 7월

스마트 스피커 작동방식(HOW)

음성인식

자연어이해

검색

자연어생성
(응답)

음성 합성

STT → 오늘 서귀포 날씨 알려줘

NLU → 오늘 서귀포 날씨 알려줘
→ Speech Act : 정보성
→ Domain : 날씨
→ Entity : 언제=오늘 , 어디=서귀포

select 날씨
where time=오늘 and location=서귀포
➔ 날씨 : 흐림, 온도 : 18도, 습도 : 30%

NLG Template
<time> <location> 날씨는 <날씨>니다. 습도는 <습도>,
기온은 <온도>로 <날씨_상황> 같아요.

오늘 서귀포 날씨는 흐립니다. 습도는 30%, 온도는
18도로 쌀쌀할 것 같아요. …
→ TTS

음성인식

자연어이해

검색

자연어생성
(응답)

음성 합성

위너의 윌리윌리 틀어줘

위너의 윌리윌리 틀어줘
→ Speech Act : Play
→ Domain : 음악
→ Enity : 가수=위너, 노래제목=윌리윌리

select 노래
where artist=위너 and song=윌리윌리

결과 없음
➔ 재생 실패
➔ 원하시는 노래를 찾을 수가 없어요

하지만

음성인식

음성인식 오류

자연어이해

검색

검색 실패

자연어생성
(응답)

음성 합성

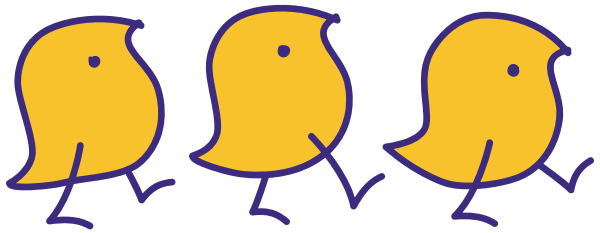
위너의 윌리윌리 틀어줘

위너의 윌리윌리 틀어줘
→ Speech Act : Play
→ Domain : 음악
→ Enity : 가수=위너, 노래제목=윌리윌리

select 노래
where artist=위너 and song=윌리윌리

결과 없음
➔ 재생 실패
➔ 원하시는 노래를 찾을 수가 없어요

오류를 교정해 보자

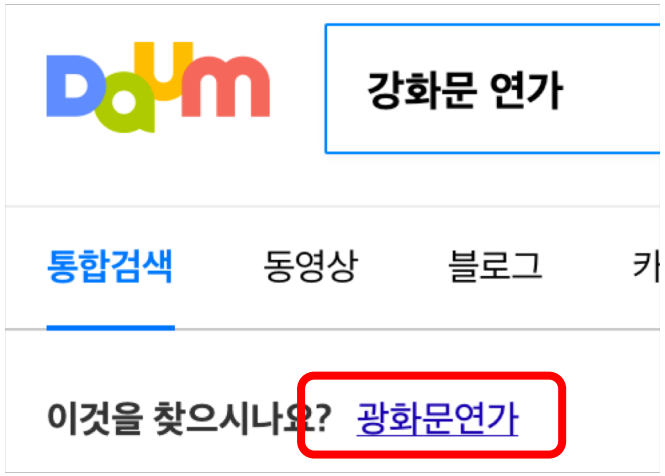


오타 교정

LangCon 2019

오타교정

- 잘못된 질의(발화)으로부터 올바른 질의(발화)로 교정하는 기술
  - 처리 순서 : 오타 판별 → 후보 검색 → 후보 검증
  - Noise Channel Model 이용
    - 교정 전 ➔ 오늘 서기호 날씨 알려줘 : 서기호 → 서귀포
    - 교정 후 ➔ 오늘 서귀포 날씨 알려줘



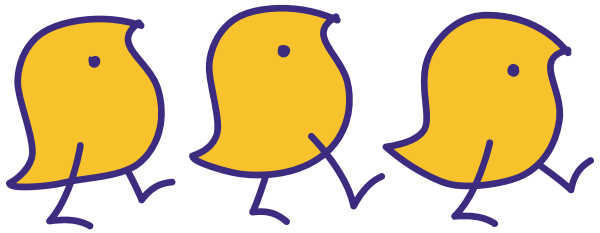
We see an observation x of a misspelled word

Find the correct word w

$$\hat{w} = \underset{w \in V}{\operatorname{argmax}} P(w \mid x)$$

$$= \underset{w \in V}{\operatorname{argmax}} \frac{P(x \mid w) P(w)}{P(x)}$$

$$= \underset{w \in V}{\operatorname{argmax}} P(x \mid w) P(w)$$

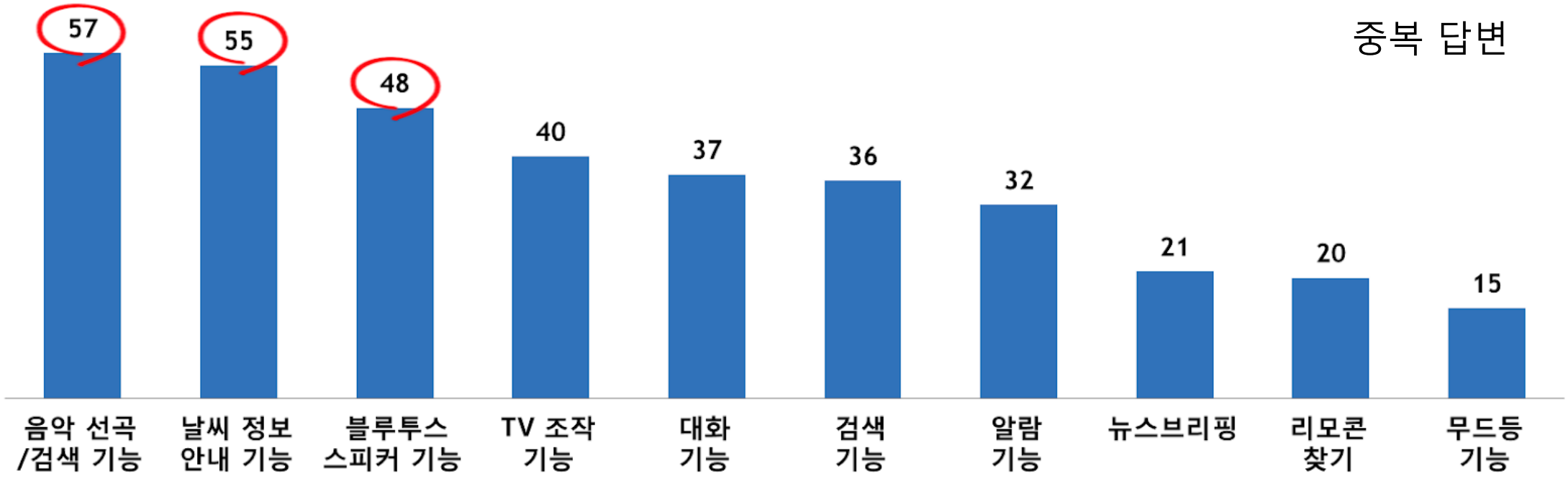


스마트 스피커에서의 음악 재생 발화 오류 교정

LangCon 2019

음악 재생 발화

- 스피커 유입 발화의 **57%** 음악 재생 발화
- **다양한 형태**로 이뤄진 가수명과 곡명으로 오류가 빈번히 발생
  - ex) iKON(가수명), 방탄소년단(가수명), 1도 없어(곡명), 花요일(곡명)



[1] "단순해서 재미 없어요"..AI 스피커 사용자 만족도 기대 이하, 한국경제, 2018.7.10

개발 과정

서비스 런칭

Grafana

로그 검토

문제점 발견

원하시는 음악을 찾을 수가 없어요
원하시는 음악을 찾을 수가 없어요
원하시는 음악을 찾을 수가 없어요
원하시는 음악을 찾을 수가 없어요
원하시는 음악을 찾을 수가 없어요
...

+ 기본적인 욕구
(내가 듣고 싶은 노래를 듣고 싶다)

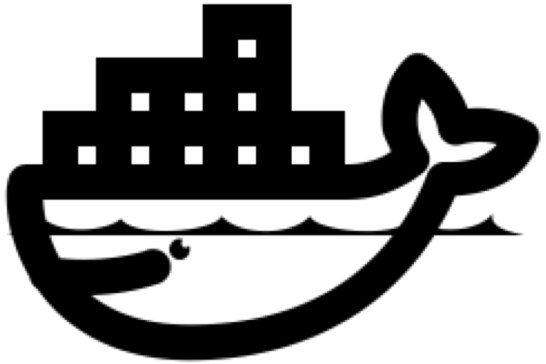
개발 과정(cont`)

토이제작

+ 호기심 1
다들 바쁨 (+나도 바쁨)
틈틈히 빠르게 만들어 보자.
- Python + open library

서비스개발

서비스용 개발을 시작
- 빠른 처리 시간
→ 내부 모듈 사용 (협업)
- 기계학습도 적용해보자
→ 학습 데이터 구축



서비스 적용

서비스 적용
- dockerizing…
런칭 후 오류 처리
- Code error
- Error Result analysis

음성 인식 오류 발생 요인

음성인식 환경

- 스마트 스피커는 보통 거실에서 사용
- 외부 유입 소리(TV)

다양한 발화자

- 가족 구성원 : 아이~할아버지

여러 도메인 지원

- 약 50개 (카카오 미니, 2018년 10월 기준)
- 음악, 택시 호출, 메세지 보내기, 날씨, 운세 등

종료 지점 예측 문제(End Point Detection)

- 음성인식이 잘리는 경우

사용자 만족도 감소

음악 재생 발화

- 음악 재생을 위한 발화
  - “아이유 음악 틀어줘”
  - “아이유의 좋은날 틀어줘” **오류 교정 대상 한정**
  - “퇴근 할 때 듣기 좋은 음악 들려줘”

음악 재생 발화 오류 유형

오류 유형	정답	인식 결과
유사 발음 오류	씨야의 바보에게 바보가	지아의 바보에게 바보가
띄어쓰기 오류	모모랜드 뿜뿜 틀어줘	모모랜드 뿜 뿜 틀어줘
영어↔한글 오류	코코 ost remember me	코코 ost 리멤버 미
	러시안 룰렛 틀어줘	Russian Roulette 틀어줘
삽입 오류	방탄소년단 run	방탄소년단 run run
삭제 오류	ABBA I Have a Dream	ABBA the dream
인지 오류	인피니티의 내꺼하자 틀어줘	샤이니의 내꺼하자 틀어줘
기호 인식 오류	76-71=	76 빼기 71 은 76 마이너스 71 은

관련 연구

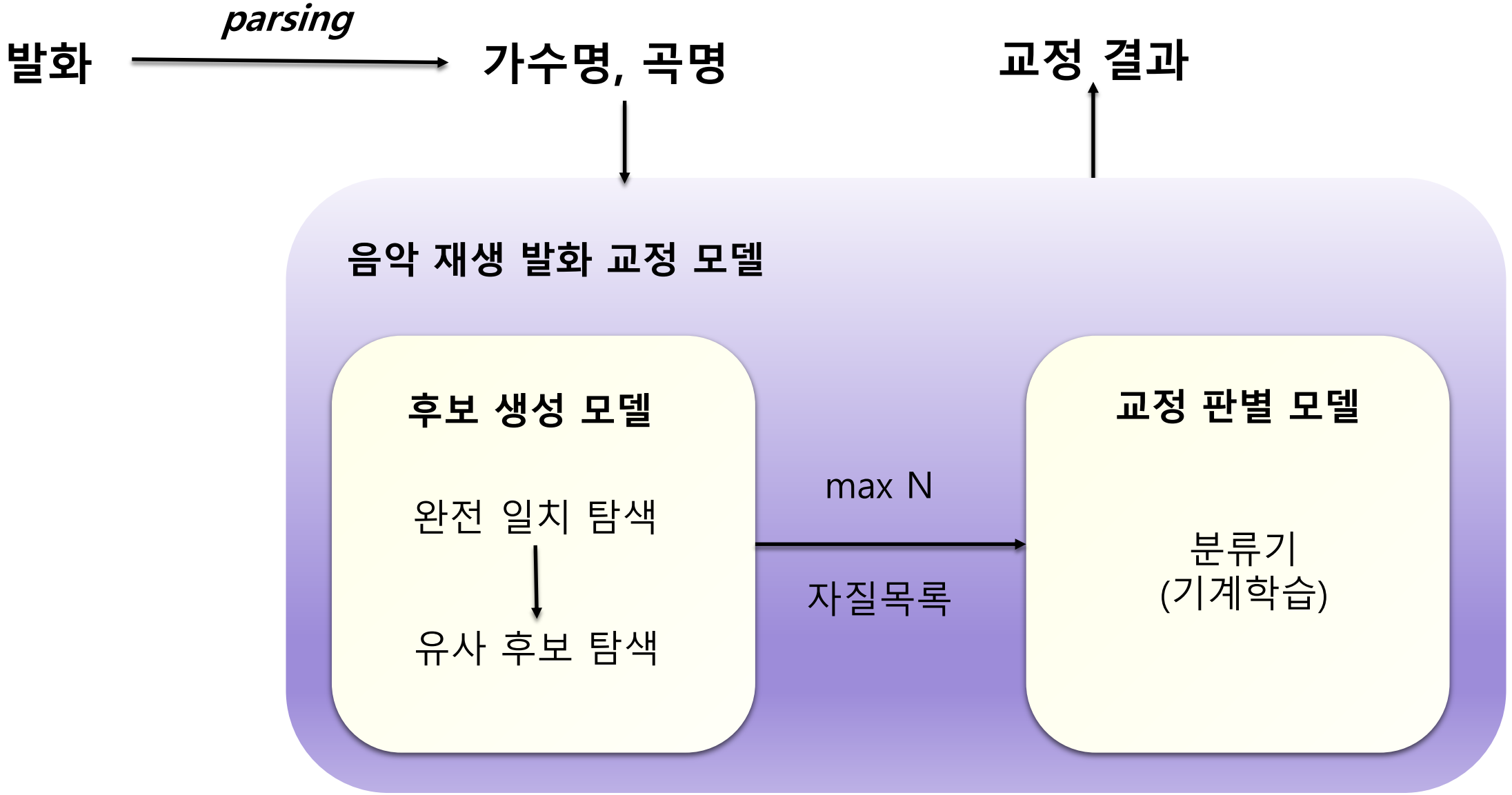
- **규칙기반 음성인식 오류교정[2]**
  - 말뭉치에서 불일치된 부분과 주변 문맥을 함께 추출하여 규칙을 학습
  - 빠른 속도로 교정 가능
  - 다른 유형의 규칙을 쉽게 학습할 수 있음

- **검색엔진(Bing)의 오타 교정 시스템을 활용한 오류 교정[3]**
  - 음성 인식된 문자열을 검색 엔진에 입력하여 교정하는 방법
  - 대량의 문서 정보를 통해 학습된 검색엔진을 활용

- **정규화 연관 거리(Normalized Relevance Distance)로 오타를 탐지하고 교정하는 방법[4]**
  - 음성인식 결과를 전이 그래프로 표현한 뒤 정규화 연관 거리를 CRF 자질로 사용하여 오타를 탐지하고 교정
  - 의미적으로 더 유사한 결과로 인식 결과를 보정

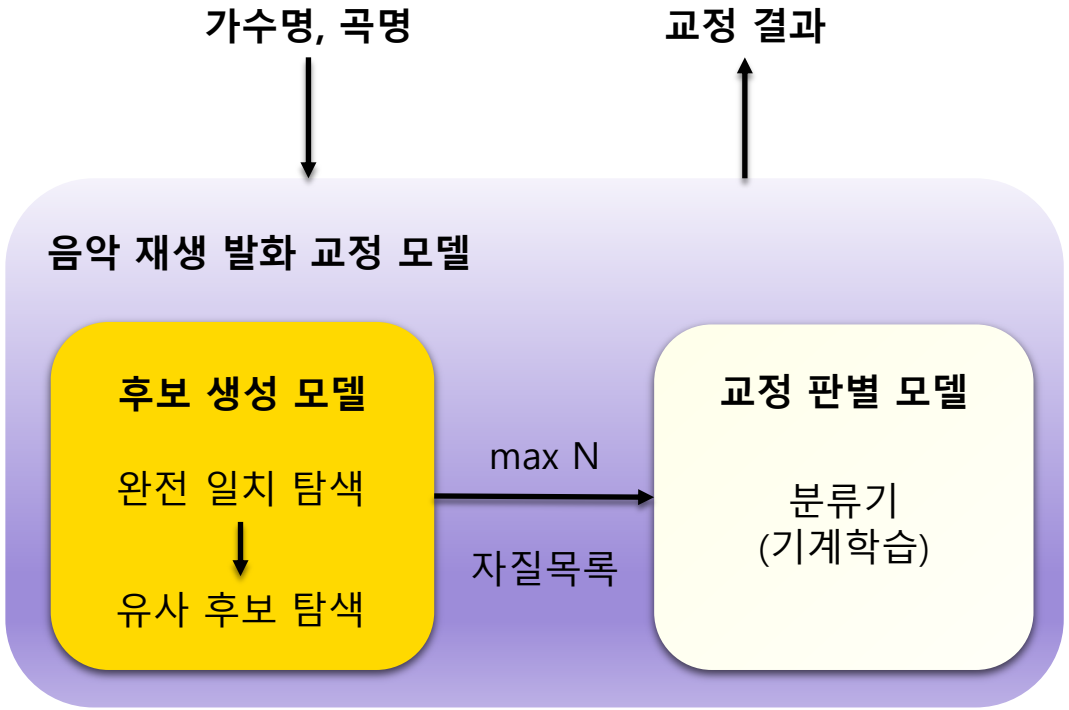
- **CRFs와 TBL을 이용한 오류 교정[5]**
  - CRF를 이용하여 오류 검출, TBL을 이용하여 오류 수정 규칙을 학습하는 방법
  - 음성 발화의 특성을 반영한 편집 거리 연산 사용

음악 재생 발화 교정 모델


parsing
발화
가수명, 곡명
교정 결과
음악 재생 발화 교정 모델
후보 생성 모델
완전 일치 탐색
유사 후보 탐색
max N
자질목록
교정 판별 모델
분류기
(기계학습)

후보 생성 모델

- 정답 후보 사전에서 입력과 **유사도가 높은 후보**를 출력하는 모델
- **탐색 방법**
  - 완전 일치 탐색: 입력과 동일한 결과가 있는지 확인
  - 유사 후보 탐색: 가수명과 곡명에 대해 유사 문자열 탐색
  - 탐색 결과 : 최대 힙(max heap)에 저장
- 유사도 계산을 위한 **전처리(pre-processing)**
  - 입력: 발음열[6]로 변환
  - 정답 후보 사전: 편집 거리가 고려된 **트라이**에 저장



정답 후보 추출을 위한 수식

문자열 간의 유사도 계산 방법을 결합하여 사용

- 편집 거리(Edit Distance), Jaro-Winkler, Overlap

$$sim\_ed(x,y) = \alpha \times sim_{ED}(x_{song}, y_{song}) + (1-\alpha) \times sim_{ED}(x_{artist}, y_{artist})$$

$$sim\_jw(x,y) = \beta \times sim_{JW}(x_{song}, y_{song}) + (1-\beta) \times sim_{JW}(x_{artist}, y_{artist})$$

$$sim\_ol(x,y) = sim_{OVERLAP}(x_{artist+song}, y_{artist+song})$$

$$sim(x,y) = \frac{sim\_ed(x,y) + sim\_jw(x,y) + sim\_ol(x,y)}{3}$$

x : 입력, y : 정답 후보

sim_ed : 곡명, 가수명에 대한 편집 거리 유사도

sim_jw : 곡명, 가수명에 대한 Jaro-Winkler 유사도

sim_ol : 전체 문자열에 대한 Overlap 유사도

sim : 최종 유사도

sim_{ED} : 편집 거리 유사도

sim_{JW} : Jaro-Winkler 유사도

$sim_{OVERLAP}$: OVERLAP 유사도

https://pypi.org/project/python-Levenshtein/

유사도 설명

- sim_{ED} : 편집 거리 유사도
  - 삽입, 삭제, 치환, 교환에 대해 두 문자열간의 편집 비용을 계산
- sim_{JW} : Jaro-Winkler 유사도
  - 두 문자열간의 교환 비용을 계산하고, prefix matching에 대한 가중치 적용
- $sim_{OVERLAP}$: OVERLAP 유사도
  - 두 문자열간의 포함 여부를 계산

예) "love my of life" , "love of my life" 의 유사도 계산

love my of life

love of my life

$sim_{ED}(love\ my\ of\ life, love\ of\ my\ life) = 0.7333$ (distance=4, length=15)

$sim_{\mathrm{JW}}(love\ my\ of\ life, love\ of\ my\ life) = 0.9555$

$sim_{OVERLAP}(love\ my\ of\ life, love\ of\ my\ life) = 1.0$ (distance=0)

유사도 설명(cont`)

sim_{ED} : 편집 거리 유사도

$$d_{a,b}(i,j) = \begin{cases} \max(i,j) & \text{if } \min(i,j) = 0, \\ \min \begin{cases} d_{a,b}(i-1,j)+1 \\ d_{a,b}(i,j-1)+1 \\ d_{a,b}(i-1,j-1)+1_{(a_i \neq b_j)} \\ d_{a,b}(i-2,j-2)+1 \end{cases} & \text{if } i,j > 1 \text{ and } a_i = b_{j-1} \text{ and } a_{i-1} = b_j \\ \min \begin{cases} d_{a,b}(i-1,j)+1 \\ d_{a,b}(i,j-1)+1 \\ d_{a,b}(i-1,j-1)+1_{(a_i \neq b_j)} \end{cases} & \text{otherwise.} \end{cases}$$

sim_{JW} : Jaro-Winkler 유사도

$$sim_j = \begin{cases} 0 & \text{if } m = 0 \\ \frac{1}{3}\left(\frac{m}{|s_1|} + \frac{m}{|s_2|} + \frac{m-t}{m}\right) & \text{otherwise} \end{cases}$$

$$sim_w = sim_j + (\ell p(1 - sim_j))$$

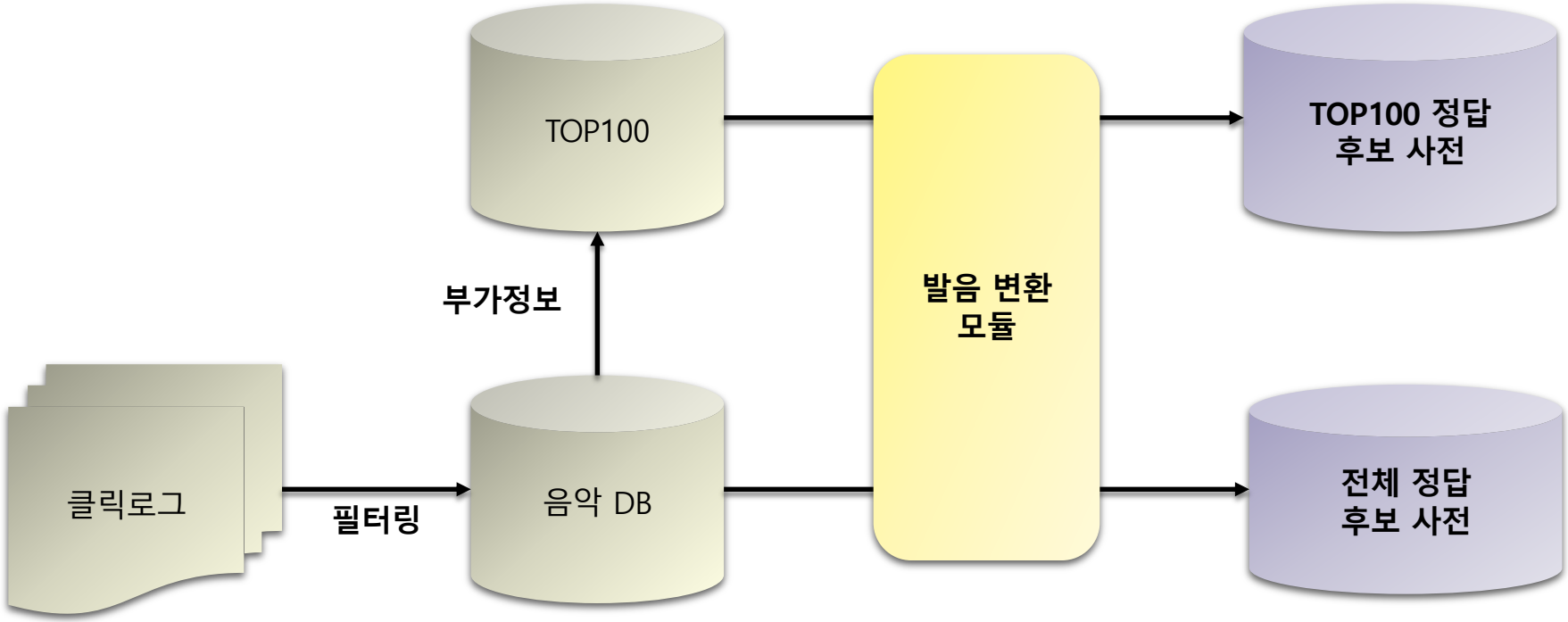
https://en.wikipedia.org/wiki/Jaro%E2%80%93Winkler_distance

$sim_{OVERLAP}$: OVERLAP 유사도

$$\text{overlap}(X,Y) = \frac{|X \cap Y|}{\min(|X|,|Y|)}$$

정답 후보 사전

- 원 데이터 ➔ 음악 DB
- 부가 정보 사용 ➔ 가수/곡 정보, 동의어
  - ex) BTS 동의어: 방탄소년단
- 정답 후보 사전 구축
  - TOP 100 정답 후보 사전 : 신곡에 대한 대응
  - 전체 정답 후보 사전 : 6개월 누적 클릭로그를 이용하여 필터링

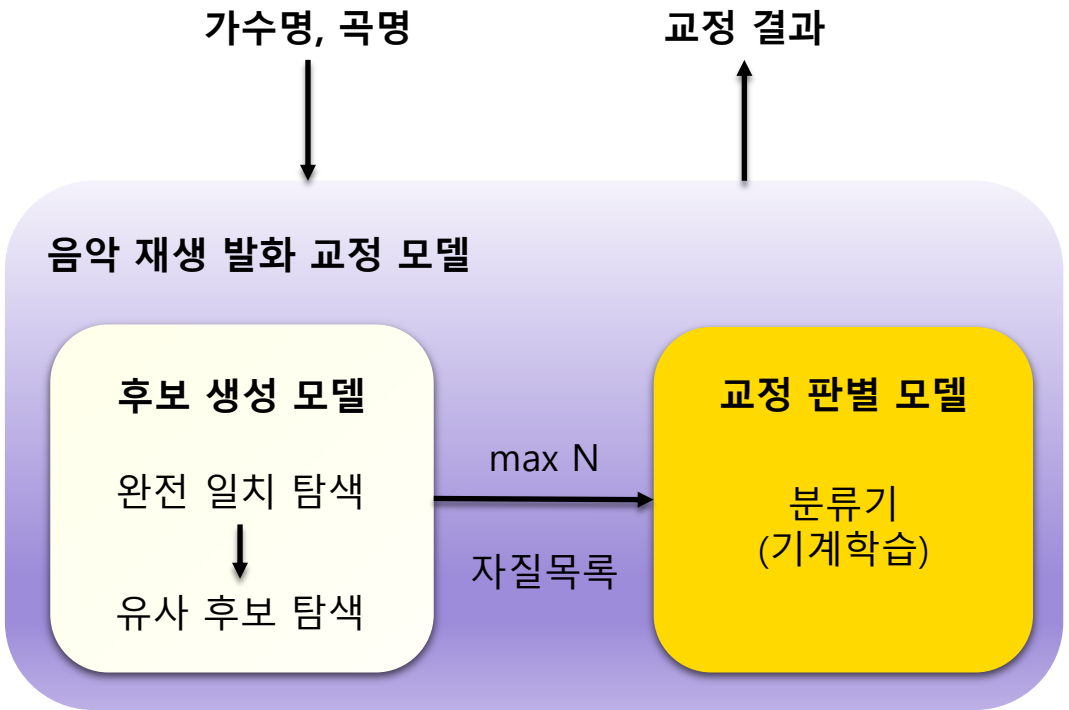


교정 판별 모델

- 후보 생성 모델에서 출력한 정답 후보들의 **교정 여부를 판별**하는 모델
  - 이진 분류 모델 : 교정 / 미교정

- **16개의 자질 정보**를 이용하여 분류
  - **가수명, 곡명 각각**에 대한 유사도 자질
    - 유사도(ed, jw) X 원형, 정규화, 역순(reverse)
  - **가수명과 곡명을 결합**한 전체 유사도 자질
    - 유사도(ed, overlap) X 원형, 정규화(normalized)

- 분류 모델
  - Random Forest(RF)를 사용
    - 앙상블 모델(Ensemble Model)
    - 의사 결정 나무(Decision Tree)를 개별 모형으로 사용



교정 판별 모델 자질

종류	자질
전체 유사도	overlap
	overlap-normalized
	edit distance
	ed-normalized
가수명 유사도	ed-artist
	ed-normalized-artist
	jaro-artist
	jaro-reverse-artist
	jaro-normalized-artist
	jaro-normalized-reverse-artist
곡명 유사도	ed-song
	ed-normalized-song
	jaro-song
	jaro-normalized-song
	jaro-reverse-song
	jaro-normalized-reverse-song

ed : edit distance

jaro : jaro-winkler

normalized : a'b of song → abofsong

reverse : abcdef → fedcba

실험 데이터

- 실제 서비스에 유입된 **로그 데이터를 사용**
  - 음악 재생 발화로 분류된 로그 데이터에서 정답 가수명, 곡명을 수동으로 부착
  - 총 9,280개 데이터
    - 교정이 필요한 경우 → 4,081개
    - 교정이 불필요한 경우 → 5,199개

발화	입력		정답	
	가수명	곡명	가수명	곡명
트와이스 wild love 틀어줘	트와이스	wild love	트와이스	What is love
blackpink의 마지막인 것처럼 틀어줘	blackpink	마지막인 것처럼	BLACKPINK	마지막처럼
위너의 윌리윌리	위너	윌리윌리	WINNER	REALLY REALLY
씨야의 snowman 틀어줘	씨야	snowman	sia	snowman
a-ha의 take me up 틀어줘	a-ha	take me up	a-ha	take me on

후보 생성 모델 실험 결과

- 유사도 임계값(minimum similarity, **ms**), 출력 후보 개수**(T)**에 따른 성능 비교

구분	ms=0.9 T=1	ms=0.6 T=1	ms=0.6 T=2	ms=0.6 T=3	ms=0.2 T=1	ms=0.2 T=2	ms=0.2 T=3
정확률 (Precision)	92.74%	88.03%	88.61%	88.73%	38.73%	37.25%	35.95%
재현율 (Recall)	5.66%	47.24%	51.60%	52.21%	53.69%	58.83%	60.65%
F1-score	10.67%	61.49%	65.22%	65.74%	45.00%	45.61%	45.14%

- 편집 거리 임계치 : 가수명 = 2, 곡명 = 6

- 평가
  - 유사도 임계값(ms)이 낮아질수록, 출력 후보의 개수(T)를 증가시킬수록
    - 재현율 증가, 정확률 감소
  - 재현율도 높이면서 정확률도 높일 수 있는 방법 강구
    - ➔ **교정 판별 모델 적용**

교정 판별 모델 실험 설명

- 후보 생성 모델의 출력 후보 이용
- 후보 생성 모델의 설정값
  - **재현율이 가장 높은 설정값** 이용
  - 유사도 임계값(ms) = 0.2, 출력 후보 개수(T) = 3
- 데이터
  - 후보 생성 모델을 이용하여 데이터 증량
    - 입력 : 9,280개
    - 출력 후보수 : 19,563개 (T=3)
    - 데이터 정제 후 개수(출력이 없는 데이터 제거) : **17,176개**
      - 교정이 필요한 경우 2,601개
      - 교정이 필요 없는 경우 14,575개

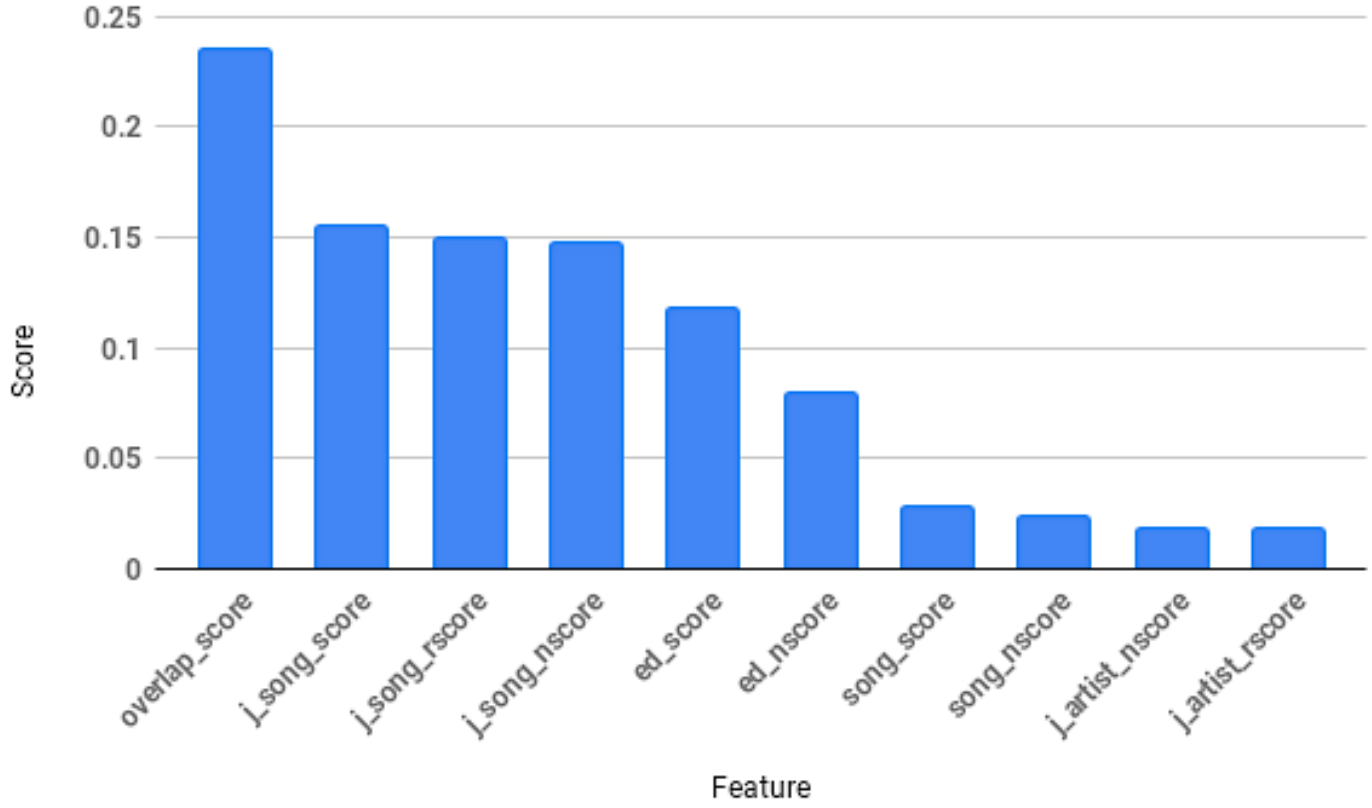
교정 판별 모델 실험 결과

- 성능 측정 방법
  - 10 fold cross validation
  - RF외 XGBoost, Logistic Regression, SVM 모델 추가 평가
  - RF
    - 나무 깊이(depth) : 40
    - 나무 수(tree) : 10, 20

Model	RF (depth=40, tree=10)	RF (depth=40, tree=20)	Logistic Regression	XGBoost (depth=40, tree=20)	SVM(rbf)
정확률 (Precision)	0.82897	0.83336	0.81959	0.80208	0.80965
재현율 (Recall)	0.77542	0.78465	0.74007	0.75852	0.78126
F1-score	0.79943	0.80634	0.77525	0.77803	0.78774

자질 중요도

- Overlap의 중요도가 높게 나옴
- 곡명과 관련된 자질들의 중요도가 높음
  - jaro-song
  - jaro-reverse-song
  - jaro-normalized-song
- 후보 생성 모델의 최대 편집 거리를 가수명은 엄격하게 설정하고, 곡명은 느슨하게 설정
- 이 설정값의 영향으로 자질 중요도가 나올수 있음
- 편집 거리 임계치 : 가수명=2, 곡명=6



결과 분석

- 음악 재생 발화에서 발생하는 오류 처리에 좋은 성능을 보임
  - 비슷한 발음으로 발생하는 오류를 효과적으로 처리

발화	입력		정답	
	가수명	곡명	가수명	곡명
존 박 니 생각 틀어줘	존 박	니 생각	존 박	네 생각
위키미키의 랄랄라 틀어줘	위키미키	랄랄라	위키미키	La La La
트와이스 처럼 틀어줘	트와이스	처럼	TWICE	CHEER UP

- 인지 오류에 대해서는 처리가 어려움
  - 가사의 일부를 곡명으로 잘못 알고 명령하는 경우

발화	입력		정답	
	가수명	곡명	가수명	곡명
빅뱅 찹쌀떡 들려줘	빅뱅	찹쌀떡	빅뱅	BAE BAE
마마무 별 헤는 밤	마마무	별 헤는 밤	마마무	별이 빛나는 밤

- **스마트 스피커에서의 오류 보정 시도**
  - 음악 재생 발화 오류를 효과적으로 처리
- 실제 서비스에서도 효과
  - 음악 재생 발화 연결 실패율 **10% 감소**
  - 음악 재생 사용률 **15% 증가**

향후연구

- 음악 재생 발화 교정 로그를 활용한 인지 오류 개선 및 에러 모델 개발
- 다국어 발음에 강건한 모델 개발
- 딥러닝을 활용한 연구
- 음악 이외의 다양한 도메인에 확대 적용

감사합니다.

Language Conference 2019

참고문헌

- [1] "단순해서 재미 없어요"..AI 스피커 사용자 만족도 기대 이하, 한국경제, 2018.7.10, https://news.v.daum.net/v/20180710162905818(2018.9.3)
- [2] 김진형, 박소영, "모바일 환경을 고려한 규칙기반 음성인식 오류 교정", 한국컴퓨터정보학회논문지, 17(10), pp. 25-33, 2012.
- [3] Youssef Bassil and Mohammad Alwani, Post-Editing Error Correction Algorithm For Speech Recognition using Bing Spelling Suggestion, IJACSA, Vol. 3, No. 2, 2012.
- [4] Yohei Fusayasu, Katsuyuki Tanaka, Tetsuya Takiguchi, Yasuo Ariki, Word-Error Correction of Continuous Speech Recognition Based on Normalized Relevance Distance, IJCAI, pp. 1257-1262, 2015.
- [5] 선충녕, 정형일, 서정연, "CRFs와 TBL을 이용한 자동화된 음성인식 후처리 방법", 정보과학회논문지 : 소프트웨어 및 응용, 37(9), pp. 706-711, 2010.

참고문헌

- [6] 국립국어원, "표준어 규정 제2부 표준 발음법", 2016.
- [7] Brill, Eric; Moore, Robert C., An Improved Error Model for Noisy Channel Spelling Correction, Proceedings of the 38th Annual Meeting on Association for Computational Linguistics, pp. 286-293, 2000.
- [8] Jaro, M. A., Advances in record linkage methodology as applied to the 1985 census of Tampa Florida, Journal of the American Statistical Association, 84(406), 414-420, 1989.
- [9] Vijaymeena, M. K.; Kavitha, K, A Survey on Similariy Measures in Text Mining, Machine Learning and Applications: An International Journal. 3(1), pp. 19-28, 2016.
- Distance
  - https://hpi.de/fileadmin/user_upload/fachgebiete/naumann/folien/SS13/DPDC/DPDC_12_Similarity.pdf
  - https://pypi.org/project/textdistance/