Smart Media Journal / Vol.7, No.2 / ISSN:2287-1322
http://dx.doi.org/10.30693/SMJ.2018.7.2.54

터치스크린 환경에서 쿼티 자판 오타 교정을 위한 n-gram 언어 모델

(N-gram based Language Model for the QWERTY Keyboard Input Errors in a Touch Screen Environment)

옹윤지*, 강승식**

(Yoon Gee Ong, Seung Shik Kang)

요 약

스마트폰과 태블릿PC 등 터치스크린을 활용한 휴대기기의 사용이 늘어나면서 데스크탑 컴퓨터나 노트북으로 수행하던 작업을 스마트폰과 태블릿PC를 이용하여 수행하는 일이 많아졌다. 그런데 휴대성을 갖춰야하는 스마트기기의 특성상, 쿼티 자판은 작은 화면 안에 조밀하게 배치된다. 그리고 이러한 점은 기계식 쿼티 자판을 사용할 때와는 다른 양상의 오타가 발생하는 원인으로 작용한다. 각 버튼이 차지하는 공간이 충분했던 기계식 쿼티 자판과 달리, 터치스크린에서의 쿼티 자판은 각 버튼에 할당되는 영역이 작아 사용자가 누르려고 의도했던 버튼이 아닌 주변의 버튼이 입력되는 경우가 자주 발생하게 된다. 본 논문에서는 어절 유니그램과 바이그램 확률을 이용한 n-gram 언어 모델 방법으로 터치스크린 환경에서 쿼티 자판으로 입력되는 문자 입력 오류를 자동으로 교정하는 방법을 제안하였다.

■ 중심어 : 터치스크린 ; 쿼티 자판 ; 오타 교정 ; N-그램 언어 모델 ; 자소 분리

Abstract

With the increasing use of touch-enabled mobile devices such as smartphones and tablet PCs, the works are done on desktop computers and smartphones, and tablet PCs perform laptops. However, due to the nature of smart devices that require portability, QWERTY keyboard is densely arranged in a small screen. This is the cause of different typographical errors when using the mechanical QWERTY keyboard. Unlike the mechanical QWERTY keyboard, which has enough space for each button, QWERTY keyboard on the touch screen often has a small area assigned to each button, so that it is often the case that the surrounding buttons are input rather than the button the user intends to press. In this paper, we propose a method to automatically correct the input errors of the QWERTY keyboard in the touch screen environment by using the n-gram language model using the word unigram and the bigram probability.

■ keywords : touch screen ; QWERTY keyboard ; error correction ; N-gram language model ; letter separation

Ⅰ. 서 론

스마트 기기는 여러 가지 강력한 기능을 갖고 있지만 휴대성의 측면에서는 장점으로 작용하는 작은 크기가 터치스크린 기반의 입력 시스템에서는 잦은 오류를 발생시키는 단점으로 작용한다[1,2]. 연구 결과에 의하면, 터치 대상의 크기가 작아질수 록 터치 오류가 급격하게 증가하는 경향을 보인다[3]. 현재 많은 터치기반의 스마트 기기들이 쿼티 자판을 사용하고 있으며, 터치스크린 환경에서의 쿼티 자판에서 발생하는 오타는 기계식 쿼티 자판에서 발생하는 오타와 주 발생 원인이 다르다[4].

점점 스마트폰과 태블릿PC 등 휴대기기의 기능이 더욱 발전하고 사용량이 늘어남에 따라 사람들은 그동안 데스크탑 PC를 사용하여 수행하던 작업을 스마트 기기를 이용하여 처리하는 경향이 강해지고 있다[5]. 직장인들뿐만 아니라 학생들 또한 과제를 수행할 때 데스크탑 PC 대신에 스마트 기기를 사용하여 작성하고 이메일로 제출하기도 한다[6]. 이렇듯 터치기반의 스마트 기기 사용률이 증가한다는 점을 생각하면, 터치스크린 환경에서의 쿼티 자판 입력 오류의 특성을 고려하여 기존의 오타 교정 방식과 차별화된 교정 방법을 개발하는 것이 필요해 보인다.

터치 기반의 스마트 기기에서 공간의 부족으로 인해 발생하는 입력 오류를 개선하기 위해 제안된 연구들을 살펴보면, 사용

* 학생회원, 건국대학교 항공우주정보시스템공학과
** 정회원, 국민대학교 소프트웨어학부
이 논문은 2017년도 정부(미래창조과학부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임 (No.NRF-2017M3C4A7068186).
또한, 이 논문은 2017년도 정부(교육부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임 (No.NRF-2017R1D1A1B03036409).

접수일자 : 2018년 03월 05일
수정일자 : 2018년 03월 28일
게재확정일 : 2018년 04월 05일
교신저자 : 강승식, sskang@kookmin.ac.kr

자 마다 손가락 크기가 다른 점을 고려하여, 감지된 정전용량에 따라 각각의 버튼 크기를 확대하거나 축소하는 비율을 조절하는 방식과 인접한 버튼을 고려하여 교정어군을 생성하고 사용자가 입력한 빈도를 기반으로 교정어를 추천해주는 방식이 존재한다[7,8]. 정전용량에 따라 버튼의 크기를 조절하는 방법은 오타율을 낮춰 줄 수는 있지만 잘못 입력된 어휘를 교정해주는 방식은 아니다. 그리고 인접한 버튼을 고려하여 교정어군을 생성하는 방식은 사용자의 사용이 많을수록 정확한 데이터가 학습되어 추천의 정확도가 올라가지만, 반대로 사용자의 사용량이 적은 사용 초기의 경우 제대로 된 교정어 추천이 어렵다는 단점이 있다. 또한 사용자의 사용량에 근거한 추천이기 때문에 문맥에 맞는 적절한 교정어인지에 대한 검증이 빠져있다.

철자 오류를 교정하는 방법은 크게 두 가지로 나뉘는데, 첫 번째는 규칙을 이용한 방법이고, 두 번째는 통계정보에 기반을 둔 방법이다[9]. 현재 오타 교정을 위해 가장 많이 사용되고 있는 방법은 규칙을 이용한 방식이지만, 이는 언어에 대한 전문적인 지식이 필요할 뿐만 아니라 규칙의 수가 늘어날수록 속도가 기하급수적으로 느려지는 단점을 갖고 있다. 또한 입력 오류로 발생하는 오타를 교정하는 데에는 교정 확률이 높지 않은 모습을 보인다. 특히, 스마트 기기로 수행하는 작업들의 경우 비정형화된 언어를 사용하는 경향이 강하기 때문에 규칙을 이용한 교정 방식보다는 통계적 방식을 사용하는 것이 더 적합하다[10]. 따라서 본 연구에서는 통계적 방식을 사용하였다. 특히, 문맥을 고려한 교정을 가능케 하기 위해 어절 단위의 n-gram 언어 모델을 사용하여 오타를 교정하도록 하였다.

본 논문에서는 인접한 버튼을 고려하여 교정어군을 생성하고, n-gram 언어모델을 이용하여 사용자의 사용량에 상관없이 문맥에 맞는 교정어를 제시할 수 있는 방법을 제안한다.

Ⅱ. 본 론

1. 어절 n-gram을 이용한 오타 교정

본 논문에서 제시하는 오타 교정 방법은 어절 단위의 n-gram 언어 모델을 사용하는데 이는 문맥을 고려한 교정을 위하여 선택한 방법이다. 그런데 말뭉치 자료에서 문장부호를 제거하지 않고 n-gram 언어 모델을 적용할 경우, 문법적 정보까지 고려되어 본래 살펴보려 했던 문맥 정보를 반영하는 데 충실하지 못하게 된다. 때문에 교정에 사용하는 말뭉치 자료에서 문장부호를 제거하는 사전 작업이 필요하다. 본 논문에서 제시하는 교정 방법의 실험에는 1차적으로 문장부호를 공백으로 대체하고 문장부호가 공백으로 대체된 말뭉치 자료에서 2개 이상의 공백은 1개의 공백으로 대체하는 작업을 수행한 말뭉치 자료를 사용하였다.

본 연구에서 제시하는 교정 방법의 단계는 그림 1과 같다.

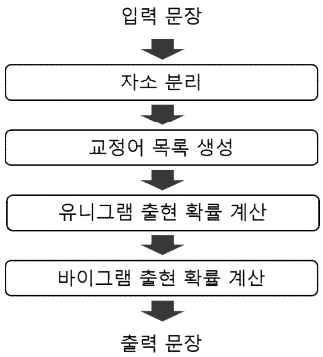


그림 1 오타 교정 과정

먼저 각 음절의 자소분리를 통하여 후보 단어 목록을 생성한다. 그 후 유니그램(unigram) 언어 모델을 이용, 후보 단어 목록 중 한 개의 후보 단어를 선택할 수 있을 경우 해당 어휘를 최종 교정어로 제시한다. 그럴 수 없을 경우, 바이그램(bigram) 언어 모델을 이용하여 문맥상 가장 적합하다고 보여지는 어휘를 최종 교정어로 제시한다.

최종 교정어를 선택하기 위해 유니그램 출현 확률 확인과 바이그램 출현 확률 확인, 이 두 가지 과정을 거치는 이유는 처리 시간을 단축하기 위함이다. 바이그램 출현 확률 확인 과정 이전에, 유니그램 출현 확률 확인을 통해 최종 교정어 선택 또는 후보 단어 목록을 줄임으로써 바이그램 출현 확률 검사 과정을 생략하거나 검사에 소요되는 시간을 줄일 수 있다.

2. 자소 교체를 통한 후보 단어 목록 생성

터치 기반의 쿼티 자판은 기기의 화면이 자판을 표시하기에 충분하지 않아 오타가 빈번하게 발생한다. 각 버튼이 차지하는 공간이 좁아 인접한 버튼을 터치하게 됨으로써 오타가 발생하는 것이다.

터치스크린 환경에서 버튼의 가로 세로 비에 따른 조작성공률을 알아본 연구 자료에 의하면 가로형 버튼의 조작 성공률이 세로형 버튼의 조작 성공률 보다 높게 나타났다[11]. 가로형 버튼은 버튼의 좌우길이가 상하길이보다 긴 형태의 버튼을 말하고, 세로형 버튼은 버튼의 상하길이가 좌우길이보다 긴 버튼을 말한다. 이는 사용자가 누르려고 의도한 버튼의 위, 아래로 인접한 버튼을 터치할 확률보다 좌, 우로 인접한 버튼을 터치할 확률이 더 높음을 의미한다.

터치스크린이 적용된 스마트 기기들의 경우 대부분 가로보다 세로가 긴 형태를 갖고 있으나 쿼티 자판은 세로보다 가로가 긴 형태를 갖고 있어 화면에 표시했을 때 각각의 버튼이 정방형이

아닌 세로의 길이가 더 긴 장방형의 형태를 갖게 된다[11]. 또한 자판의 높이는 조절이 가능하고 이를 통해 오타율을 감소시킬 수 있지만[12], 자판의 좌우 길이는 화면크기의 제약을 받기 때문에 공간부족으로 인한 오타발생을 줄이는 것이 어렵다.

본 연구에서는 입력된 글자를 자소 분리한 후 후보 단어 목록을 생성할 때 입력된 버튼의 좌우로 인접한 버튼에 해당하는 자음, 모음만을 후보로 삼았다. 예를 들어, “먹었다”를 입력하려 했으나 “먹았다”가 입력되었을 때 그림2와 같이 자소 분리를 한 후, 각 자음과 모음에 대응되는 좌우 인접 버튼의 자음과 모음을 후보로 삼는다.

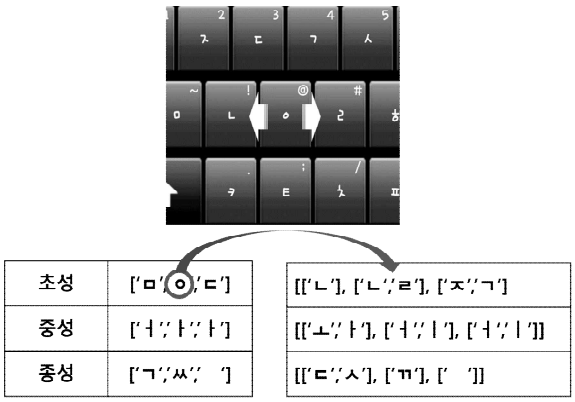


그림 2 자소 분리와 후보 목록 생성

이렇게 얻은 각 글자의 초, 중, 종성의 교정 후보 초, 중, 종성을 그림 3과 같이 조합하여 후보 음절들을 얻는다. 이때 초, 중, 종성 중 오타는 최대 한 개 존재한다고 가정한다.

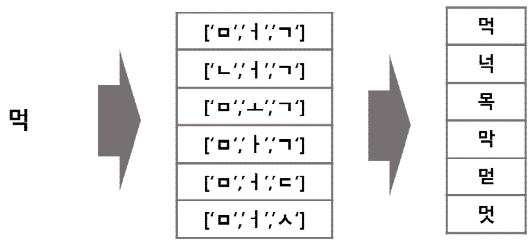


그림 3 교정 후보 초, 중, 종성의 결합

예를 들어, ‘먹’이라는 음절에 대하여 후보 음절을 얻는 과정을 살펴보면 먼저 초, 중, 종성을 분리하여 ‘ㅁ’, ‘ㅓ’, ‘ㄱ’을 얻고, ‘ㅁ’에 대한 교정 후보로 ‘ㅁ’과 인접한 ‘ㄴ’을 얻는다. 마찬가지 방식으로 ‘ㅓ’에 대한 후보로 ‘ㅗ’와 ‘ㅏ’를 얻고, ‘ㄱ’에 대한 후보로 ‘ㄷ’과 ‘ㅅ’을 얻는다. 그 다음, 초성이 오타라고 가정했을 경우 초성인 ‘ㅁ’에 대한 후보인 ‘ㄴ’을 초성으로 지정하고 나머지 중성과 종성은 입력된 ‘ㅓ’와 ‘ㄱ’으로 유지하여 ‘넉’이라는 후보 음절을 얻는다. 같은 방식으로 중성이 오타라고 가정했을 경우 후보 음절로 ‘목’, ‘막’을 얻고 종성이 오타라고 가정했을 경우 후보 음절로 ‘멀’, ‘멋’을 얻는다. 그리고 오타가 없을 경우를 고려하기 위하여 ‘먹’ 또한 후보 음절 목록에 포함시킨다.

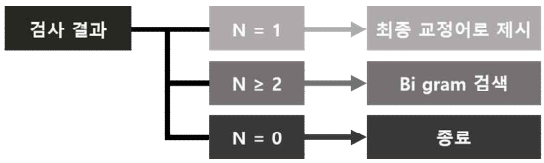
이 과정을 모든 음절에 대하여 수행하면 표 1과 같이 각 음절에 해당하는 후보 음절들을 얻게 된다. 이 후보 음절들을 조합하여 “먹았다”, “먹랐다”, “먹잤다”, “먹었다” 등 과 같은 교정어들을 생성한 후, 이 교정어들과 입력된 단어를 포함하여 후보 단어 목록을 생성한다.

표 1. 후보 음절 목록

‘먹’의 교정 후보	‘았’의 교정 후보	‘다’의 교정 후보
넉	랐	자
목	잤	가
막	었	더
먿	있	디
멋	앞	

3. 유니그램 정보를 이용한 후보 단어 선택

바이그램 언어 모델을 사용하여 출현 확률을 검사하기 전에, 사전에 준비해둔 말뭉치 자료의 어절단위 유니그램 출현 빈도 자료에서 후보 단어 각각의 출현 확률을 검사한다. 유니그램 출현 빈도 자료는 각 어절을 key로, 그리고 해당 어절의 출현 횟수를 value로 하여 생성한 dictionary형 자료이다. 그리고 검사 결과에 따라 그림 4와 같이 후보 단어 중 한 단어를 최종 교정어로 제시하거나 최종 교정어 제시를 보류하고 후보 목록을 수정한다.



N = Uni gram 빈도 수가 0이 아닌 교정어 개수

그림 4 유니그램 출현 빈도 검사 결과에 따른 선택

모든 후보 단어의 출현 빈도가 0일 경우에는 모든 후보 단어가 말뭉치 자료에 존재하지 않는다는 의미이므로, 바이그램 출현 확률 검사 결과도 모든 후보 단어의 출현 확률이 0%로 나올 것이다. 따라서 최종 교정어 제시를 하지 않고 교정 과정을 종료한다.

그리고 출현 빈도가 0이 아닌 후보 단어가 1개뿐인 경우도 마찬가지로 바이그램 출현 확률 검사 결과가 유니그램 출현 빈도 검사 결과와 같이 1개의 후보 단어만이 가능성 있는 교정어로 제시될 것이므로 바이그램 출현 확률을 검사하는 과정을 생략

하고 해당 후보 단어를 최종 교정어로 제시한다.

마지막으로 출현 빈도가 0이 아닌 후보 단어가 2개 이상일 경우에는, 후보 단어 목록 중 출현 빈도가 0인 후보 단어들을 목록에서 제외한 후 수정된 후보 단어 목록을 다음 과정인 바이그램 출현 확률 검사에 사용하도록 한다.

4. 바이그램 정보를 이용한 후보 단어 선택

유니그램 출현 빈도 자료를 어절 단위로 생성했던 것과 마찬가지로 바이그램 출현 빈도 자료 또한 문맥을 반영하기 위해 어절 단위의 바이그램을 만들고 각 바이그램을 key로, 해당 바이그램의 출현 빈도를 value로 하여 dictionary형 자료를 생성해 둔다.

앞서 유니그램 출현 빈도 검사 과정을 거쳐 수정된 후보 단어 목록의 단어들에 대해 교정대상 어절 이전에 입력된 어절과 함께 묶어 바이그램을 만들고, 만들어둔 바이그램 출현 빈도 자료를 사용하여 이 바이그램의 출현 확률을 수식 1과 같이 계산한다. 그리고 이렇게 얻은 후보 단어들의 바이그램 출현 확률을 비교하여 가장 출현 확률이 높은 후보 단어를 최종 교정어로 제시한다. 만약, 입력한 단어와 교정어의 출현 확률이 같고 그 값이 다른 후보들과 비교하여 가장 높을 경우, 입력한 단어를 유지하도록 한다.

$$P(w_j \mid w_i) = \frac{P(w_i, w_j)}{P(w_i)} = \frac{C(w_i, w_j)}{\sum_w C(w_i, w)} = \frac{C(w_i, w_j)}{C(w_i)} \quad (1)$$

w_i : 교정 대상 어절 이전에 입력된 어절
w_j : 교정어
$C(w_i, w_j) = Bigram\ \ w_i, w_j$의 출현 빈도
$P(w_i, w_j) = Bigram\ \ w_i, w_j$의 출현 확률

예를 들어, “나는 아침에 밥을 먹았다”라고 입력을 하였고 현재 검토 중인 어절이 “먹았다”일 때 “먹았다”의 후보 단어 중 “막았다”와 “먹었다”만 유니그램 출현 빈도가 0이 아니라고 하자. 바이그램 출현 확률은 유니그램 출현 빈도가 0이 아닌 후보 단어들에 대해서만 계산하면 되므로, “밥을 막았다”와 “밥을 먹었다”의 바이그램 출현 확률을 계산하면 된다. “밥을 먹었다”라는 표현은 자주 쓰이는 반면, “밥을 막았다”라는 표현은 일반적으로 쓰이지 않는 표현이다. 그러므로 “밥을 먹었다”의 출현 확률이 더 높게 계산될 것이다. 후보 단어는 “먹었다”와 “막았다”만 남아 있었으므로 바이그램 출현 확률이 가장 높은 “먹었다”를 입력된 “먹았다”의 최종 교정어로 제시한다.

5. 실험 및 결과

가. 실험 환경

실험에 사용된 말뭉치 자료는 한국과학기술정보연구원에서 정보검색 시스템 평가를 위해 구축한 한글 테스트 컬렉션을 사용하였다.* 이 말뭉치에 포함된 한글 텍스트 데이터는 신문뉴스 기사 내용으로 시스템 평가를 위해 구축했기 때문에 비교적 잘 정제된 자료이다. 말뭉치의 크기는 11,745,281개의 어절로 구성되어 있으며, 이 자료에서 추출된 유니그램은 996,600개, 바이그램은 4,219,136개이다.

유니그램과 바이그램 출현 확률을 이용한 오타 교정율 비교에는 말뭉치 자료 중 임의로 선택한 문장들을 좌우인접버튼을 잘못 눌렀다는 가정 하에 글자를 변형하여 오타를 만든 데이터와 스마트폰으로 직접 타이핑한 데이터를 사용하였으며, 이 문장들은 총 1,902개의 단어로 구성되어 있다. 마찬가지로 유니그램 출현 빈도 검사 과정 유무에 따른 소요시간 비교에도 말뭉치 자료 중 임의로 선택한 문장을 스마트폰으로 입력하여 사용하였다.

한 어절 당 오타율은 30% 미만으로 가정하였고 계산된 오타가 존재하는 음절 개수는 소수점 이하 버림 처리하였다. 이 경우 한 어절 당 음절 개수가 3개 이하일 때에는 오타율이 30% 미만으로 설정되어 있으므로 오타가 존재하는 음절 개수는 0.9개 미만으로 계산되고 소수점 이하를 버림 처리하면 0개가 되어 오타가 없는 것으로 고려되므로 이를 보완하기 위해 어절 당 최소 1개의 오타가 존재한다고 가정하고 교정을 수행하도록 하였다. 예를 들어, “국민건강생활지침”과 같은 단어는 1개 또는 2개의 오타가 있다고 가정하여 교정어를 생성하고, “국민”과 같은 단어는 1개의 오타가 있다고 가정하여 교정어를 생성한다.

나. n-gram 언어 모델을 이용한 오타 교정

바이그램 출현 확률 검사 과정을 통해 오타가 문맥에 맞게 교정되었는지를 확인해보기 위하여 단순히 유니그램 출현 빈도 검사만을 거쳐 빈도수가 가장 큰 후보 단어를 최종 교정어로 제시하는 방법을 사용했을 경우의 오타 교정 결과와 유니그램 출현 빈도 검사 과정과 바이그램 출현 확률 검사 과정을 함께 사용하는 방법으로 교정했을 때의 오타 교정 결과를 비교해 보았다.

* http://www.kristalinfo.com/TestCollections/#hkib

표 2. 오타 교정 결과 (자동 생성 데이터)

	유니그램	유니그램 + 바이그램
오타가 포함된 어절 수	190	190
올바르게 교정된 어절 수	88	188
교정되지 않은 어절 수	102	2
틀리게 교정된 어절 수	0	5

표 3. 오타 교정 결과 (직접 입력한 데이터)

	유니그램	유니그램 + 바이그램
오타가 포함된 어절 수	418	418
올바르게 교정된 어절 수	352	406
교정되지 않은 어절 수	66	12
틀리게 교정된 어절 수	3	6

그 결과, 자동으로 생성한 데이터를 사용하여 진행한 실험에서는 유니그램 출현 빈도를 이용하여 오타를 교정했을 경우 올바르게 교정된 경우는 오타가 존재하는 어절 190개 중 88개였지만 바이그램 출현 확률을 추가하여 교정한 경우에는 190개 중 188개가 올바르게 교정되었으며 교정되지 않은 경우는 2개였다. 직접 입력한 데이터를 사용하여 진행한 실험에서는 유니그램 출현 빈도 검사 과정만 거쳐 오타를 교정했을 경우, 오타가 존재하는 어절이 올바르게 교정된 경우는 오타가 존재하는 어절 418개 중 352개 였고, 바이그램 출현 확률 검사 과정을 추가하여 교정한 경우에는 유니그램 출현 빈도 검사 과정만 거쳐 오타를 교정했을 때 교정되지 않은 54개의 어절이 추가로 올바르게 교정되었다.

교정 내용 중 한 가지를 예로 살펴보면, “시행착오를 고치며”라는 문구를 유니그램 출현 빈도 검사만을 사용하여 교정하였을 경우, “고치며”라는 어휘 자체에는 오류가 없고 “거치며”라는 어휘보다 출현 빈도수가 더 많아 오타 교정이 되지 않았지만, 바이그램 출현 확률을 검사했을 경우에는 “시행착오를 고치며” 보다는 “시행착오를 거치며”라는 문구의 출현 확률이 더 높기 때문에 문맥에 맞게 “고치며”가 “거치며”로 교정된 것을 확인할 수 있었다.

실험을 통해 유니그램 출현 빈도 검사 과정만을 거쳐 오타를 교정했을 때 보다 바이그램 출현 확률 검사 과정을 추가했을 때 오타 교정률이 자동 생성 데이터를 사용했을 경우 46.32%에서 98.95%로 증가하였으며 직접 입력하여 제작한 데이터를 사용했을 경우 84.21%에서 97.13%로 증가하였다.

좌우 인접버튼을 잘못 터치되었다는 가정 하에 자동으로 생성한 데이터를 사용했을 때에는 교정률이 98.95%이지만 직접 입력하여 만든 데이터를 사용했을 때에는 교정률이 97.13%로 떨어진 것을 확인할 수 있다. 이는 실제로 입력하는 경우에 좌우 인접버튼이 아닌 상하 인접버튼을 누르는 경우나 아예 다른 버튼을 잘못 누르는 등의 원인으로 인한 오타가 존재하기 때문이다.

또한 직접 입력하여 제작한 데이터를 사용하여 교정을 진행한 경우, 오타가 존재하지 않음에도 교정되어 오히려 오류가 생긴 경우가 존재했다. 유니그램 출현 빈도 검사 과정만 거쳤을 경우 교정 대상이었던 1,902개의 어절 중 약 0.16%에 해당하는 3개 어절에서 교정 오류가 발생하였으며, 바이그램 출현 확률 검사 과정을 추가하여 교정했을 경우 약 0.32%에 해당하는 6개 어절에서 교정 오류가 발생하였다. 유니그램 출현 빈도 검사 과정만 거친 경우, 문맥이 고려되지 않아 오류가 생긴 것으로 이는 바이그램 출현 확률 검사과정을 추가하며 교정되었다. 그러나 바이그램 출현 확률 검사 과정을 추가한 후에도 교정 오류가 생긴 것은 바이그램의 특성 상, 교정의 대상이 되는 어절의 앞과 뒤를 모두 고려하여 교정하지 못하는 점 때문이며, 추후 트라이그램(trigram)을 사용하여 보완이 가능할 것이다.

다. 유니그램 정보 사용에 따른 소요시간 비교

유니그램 출현 빈도 검사 과정을 바이그램 출현 확률 검사 과정 이전에 추가함으로써 오류 교정 시간을 단축시킬 수 있다. 의도에 맞게 소요시간이 단축되었는지 확인해보기 위해 교정 과정에서 유니그램 출현 빈도 검사 과정을 거치지 않고 바이그램 출현 확률 검사 과정만을 거쳐 교정된 시간과 유니그램 출현 빈도 검사 과정을 거친 후 바이그램 출현 확률 검사를 하여 교정했을 때 소요된 시간을 비교해 보았다. 소요 시간을 측정하는 데에는 파이썬에서 제공하는 라이브러리 중 하나인 timeit을 사용하였다. 교정 작업이 시작되는 부분과 끝나는 부분에 default_timer() 함수를 이용하여 각 시점의 performance counter 값을 받았으며 두 값의 차를 계산하여 소요 시간을 얻었다. 그 결과, 유니그램 출현 빈도 검사 과정을 거치지 않고 바이그램 출현 확률 검사 과정만을 거쳐 교정했을 때 3,717초가 소요되었는데 비해 유니그램 검사 과정을 거쳤을 때는 117초로 실행 시간이 96.85% 단축되었다.

III. 결 론

터치스크린 환경에서 쿼티 자판을 사용하여 문자를 입력할 때 공간의 부족으로 인해 오타가 빈번히 발생하는 문제점을 해결하기 위하여 인접 버튼을 오류 교정 후보로 가정하고 문맥에 맞게 오타를 교정하기 위하여 n-gram 언어 모델을 이용한 오타 교정 방법을 제안하였다. 바이그램 언어 모델을 사용하여 문맥에 맞는 교정어를 제시함으로써 유니그램 출현 빈도만을 이용하여 교정했을 때보다 더 높은 교정률을 얻어내는 데 성공했으며, 바이그램 출현 확률을 확인하는 과정 전에 유니그램 출현 빈도 검사 과정을 추가하여 교정에 소요되는 시간을 단축시켰다.

REFERENCES

[1] 김호식, 전재웅, 최윤철, “터치스크린 기반 스마트폰에서의 한글 입력 기법,” *한국정보과학회 학술발표논문집*, Vol.38, No.1, 263-265쪽, 2011년 6월.

[2] 박경민, 최훈, 이창건, 황인태, 이칠우, ""휴대단말을 위한 지능형 사용자 인터페이스 플랫폼," *스마트미디어저널*, 제1권, 제4호, 44-51쪽, 2012년 12월

[3] N. Henze, E. Rukzio, and S. Boll, “100,000,000 Taps : Analysis and Improvement of Touch Performance in the Large,” *Proc. of the 13th Int. Conf. on Human Computer Interaction with Mobile Devices and Services*, Stockholm, Sweden, pp. 133-142, Aug. 2011.

[4] 최철, 박세진, 김철중, 권규식, “쿼르타이 키보드에 기초한 인간공학키보드설계를 위한 오타율분석,” *대한인간공학회 학술대회논문집*, 142-145쪽, 2000년 4월.

[5] M. Sarwar and T. R. Soomro, “Impact of Smartphone’s on Society,” *European J. of Scientific Research*, vol. 98, no. 2, pp. 216-226, Mar. 2013.

[6] A. Lahmati, and L. Zhong, “Studying Smartphone Usage: Lessons from a Four-Month Field Study,” *IEEE Trans. Mobile Computing*, vol. 12, issue 7, pp. 1417-1427, July 2013.

[7] 정유선, 최동민, “스마트 기기 사용자 적응형 가변 키보드,” *한국전자통신학회 논문지*, 제 12권, 제 6호, 1167-1172쪽, 2017년 12월

[8] 권오상, “터치스크린 기반 스마트폰에서 키보드 인접 오타 정정 방식을 이용한 입력어 추천 시스템 구축,” *한국정보과학회 학술발표논문집*, 39권, 1호, 67-69쪽, 2012년 6월.

[9] 박승현, 이은지, 김판구, ""한글 편집거리 알고리즘을 이용한 한국어 철자오류 교정방법,"" *스마트미디어저널*, 제6권, 제1호, 16-21쪽, 2017년 3월

[10] 김민호, 권혁철, 최성기, “어절 n-gram을 이용한 문맥의존 철자오류 교정,” *정보과학학회논문지*, 제41권, 제12호, 1081-1089쪽, 2014년 12월

[11] B. Koo and K. Chung, “Accuracy based on the Widths of the Buttons on Smartphone Touchscreens,” *J. of Korean Society of Design Science*, vol. 26, no. 2, pp. 127-143, May 2013.

[12] 한성숙, 최진해, 홍지영, 오의택, 김수민, 전현주, “손 크기에 따른 운지 범위를 고려한 맞춤형 터치 키보드의 사용성 개선효과에 관한 연구,” *대한인간공학회 학술대회논문집*, 196-200쪽, 2014년 11월.

저 자 소 개

옹윤지(학생회원)

2017년 건국대학교 항공우주정보시스템공학과 학사 졸업(공학사).

<주관심분야 : 자동제어시스템, 자연어처리, 철자오류교정>

강승식(정회원)

1986년 서울대학교 전자계산기공학과 학사 졸업.

1988년 서울대학교 전자계산기공학과 학과 석사 졸업.

1993년 서울대학교 전자계산기공학과 학과 박사 졸업.

<주관심분야 : 자연어처리, 텍스트마이닝, 빅데이터 분석, 상황인지 컴퓨팅>