

세벌식을 응용한 효율적인 한글 입력의 제안

한국과학기술원 전산학전공
김 용 목 · 김 진 형

The Proposal of Effective Hangeul Input Based on Sebeolsik(Three-Layered) Key Layout

Yongmook Kim, Jin Hyung Kim
Division of Computer Science, KAIST, Daejeon, Korea

요 약

현재 우리나라의 표준 자판은 두벌식이며, 한글 입력기에 대한 연구도 두벌식을 중심으로 되어 왔다. 물론 세벌식도 있지만 널리 보급되지 못한 상태이며, 연구를 한다고 해도 글쇠배열 쪽으로 편향해 있지, 세벌식 구조 자체에 대한 평가와 연구가 제대로 이루어지지 못한 실정이다. 본 연구에서는 <날개셋> 한글 입력기라는 새로운 입력기를 개발하여, 세벌식 글쇠배열을 응용한 효율적인 한글 입력 방식을 제안한다. 기존 연구들이 한국어와 한글 맞춤법의 특성을 살린 것이거나 세벌식과 두벌식을 절충하는 것이라면, 본 연구는 세벌식 자체의 장점을 살려 입력기의 내부 기능을 향상시켰다. 그래서 세벌식의 당위성을 단순한 글쇠 배열 차원의 능률이 아닌, 구조적이고 원천적인 관점에서 역설했다. 이들 고급 기능들은 초/중/종성을 순서 없이 쳐도 조합이 성립될 수 있는 세벌식 모아치기 원리에 근거를 두고 있다.

인지과학적 신경망

PC가 보급되면서 키보드를 통한 문자 정보의 입력량이 많아졌다. 이에 따라 컴퓨터 자판에서의 효율적인 문자 입력 방법이 요구되고, 각국에서는 이에 대한 연구가 활발히 진행되고 있다. 자기 나라 언어나 정서법을 분석하기도 하고, 자주 내는 오타 패턴을 분석하여 바로잡아 주는 기능을 넣는다. 소리글자를 최대한 지능적으로 정확하게 한자로 바꾸는 방법을 연구하기도 하며 속기라든가 음성 인식, 필기 인식도 주요 연구 대상이 되기도 한다.

한글은 자모 수가 적고 과학적인 체계를 갖추고 있어 한자나 가나 같은 외국 문자보다 쉽고 간편하게 입력할 수 있다. 그러나 한글의 과학적인 원리를 응용하여 쉽고 자연스런 입력이 가능함에도 불구하고 한글 자판에 대한 공식적인 연구는 기존하는 두벌식 자판 이후 별다른 진전이 없다. 한글 창제 원리와 글자 조합 원리에 부합하는 세벌식이 제안돼 있으나 일반인들에 대한 홍보가 부족하여 널리

보급되어 있지 못한 실정이다.

세벌식을 이용하면 자모의 삭제, 수정 기능에서 현재의 두벌식 입력 체계가 구현할 수 없는 여러 새로운 기능들을 구현할 수 있다. 본 연구에서는 <날개셋> 한글 입력기라는 새로운 한글 입력기를 소개함으로써, 세벌식에서만 가능한 효율적인 한글 입력 방법에 대해 제안하려 한다.

연구 방향과 기존 사례 분석

현재 국내에 보급된 모든 컴퓨터에 인쇄되어 있는 글자판은 자음 한 벌, 모음 한 벌이 들어간 두벌식 자판이다. 두벌식은 글쇠 수가 적고 기호와 숫자 배열이 영문 자판과 호환된다는 장점은 있지만, 초성과 종성을 구분할 수 없어 한글의 구성 원리와 어긋난다는 비판을 받고 있다. 그래서 두벌식 자판을 쓰면 기계식 타자기에서는 받침이 들어갈 때마다 Shift를 눌러야 한다. 전자식 타자기를 쓰면 그나마 전자 칩에서 오토마타가 구현되는데, 조합 중인 글자가 바로 나타나지 않아 심리적으로 매우 불편하다. 이것이 컴퓨터에 와서는 도깨비불 현상으로 바뀌었다.

반면 세벌식은 한글을 이루는 초성, 중성, 종성을 모두 글자판에 구현한 구조로, 두벌식과는 달리 한글 구성 원리에 맞고 앞에서 언급한 문제가 없으며, 초, 중, 종성 한글의 어떤 낱자라도 바로 입력할 수 있다는 큰 장점을 가진다. 그러나 많은 사람들이 이 장점을 오히려 글쇠 수가 너무 많다는 단점으로만 인식하고 있어 세벌식으로의 전환이 활발하게 이루어지지 않아 왔다.

세벌식은 모든 한글 성분이 글쇠로 따로 배당되어 있기 때문에 입력기를 구현하는데 별도의 오토마타가 필요 없다. 그래서 타자기에 관한 한 세벌식은 두벌식에 대해 절대적인 우위에 있다. 그럼 컴퓨터에서는 두벌식과 세벌식의 차이가 줄어들었는가 하면 그렇지 않다. 두벌식 입력기는 컴퓨터에서야 겨우 구현되지만, 세벌식 입력기는 기본적으로 구현돼 있고, 컴퓨터의 도움을 받으면 그 원리를 응용하여 아무 낱자나 바로 입력하고 고치고 지우는 것 같은 여러 편리한 입력 기능을 자연스럽게 직관적으로 구현할 수 있기 때문이다.

학술적으로나 민간적으로 세벌식을 중심으로 순수하게 한글 입력기를 연구한 것을 보면 공 병우 세벌식 글쇠배열을 나름대로 개선하거나,¹⁾ 왼손에 중성과 종성을 중첩 배당하여 4단 자리를 쓰지 않은 세벌식을 구현하거나,²⁾ 오른손으로 타자를 시작하면 세벌식, 왼손으로 타자를 시작하면 두벌식을 자동 인식하게 하는 것³⁾ 등이 있다. 최근에 발표된 안 마태 동시 입력 방식⁴⁾도 결국 이 분야의 연구라고 볼 수 있다.

물론 이런 사례들은 한글을 낫게 만들려고 노력하는 사람들이 내놓은 결과임이 틀림없다. 하지만 이들은 세벌식을 표준 두벌식 자판과 절충시키거나, 어떻게든 세벌식의 단점, 특히 글쇠 수를 줄여 보자는 연구에 가깝지 세벌식 자판 자체의 특성과 장점을 살린 연구가 아니라는 점에서 한계가 있다.

한글 입력기의 구성 요소

로마자 입력기를 설계하는데 필요한 구성 요소는 Qwerty 배열이나 Dvorak 배열이나 하는 것처럼 글쇠배열이 전부이다. 그러나 한글 입력기는 어떤 환경에서 쓰이는 어떤 방식의 입력기를 만들든, 최소한 다음과 같은 네 가지 구성 요소를 가진다. 키보드 글쇠 수보다 더 많은 문자를 입력하면서 여러 가지 부가적인 규칙을 정해 주어야 하기 때문이다.

1. 글쇠배열

컴퓨터용 한글 입력기에서 글쇠배열이란, 공백을 제외한

47개 글쇠에 Shift를 포함하여 94개 자리에 배당되는 문자들의 집합이다. 한글 글쇠배열은 두벌식이나 세벌식이나를 규정하는 정보를 갖고 있으며, “세벌식 390”, “표준 두벌식”처럼 고유한 이름도 있다. 대개는 이 글쇠배열의 이름이 이 한글 입력 방식의 이름이 된다.

글쇠는 지정된 글자나 한글 낱자를 입력하는 글쇠일 수도 있고, 입력기가 정한 특수한 기능을 수행하는 글쇠일 수도 있다. 예를 들어 두벌식 옛한글 입력기에는 음절 구분을 위해 한글 조합을 강제로 중단시키는 글쇠가 특수 글쇠의 예라고 하겠다.

2. 낱자 결합 규칙

ㄴ 다음에 ㄱ가 오면 ㄴ가 되고, 받침 ㄴ 다음에 ㅎ이 오면 ㄴㅎ이 되게 하는 규칙을 낱자 결합 규칙이라고 한다. 옛한글 입력기라면 글쇠 개수보다 훨씬 많은 옛한글 낱자를 입력해야 하므로 대단히 많은 조합이 필요할 것이다. 반면, 현대 한글만 입력하면 되고, 세벌식 최종 글쇠배열을 쓰고 있다면 받침에 대해 낱자 결합 규칙을 전혀 정할 필요가 없다. 세벌식 최종은 모든 접받침이 글쇠에 배당돼 있기 때문이다.

이 규칙을 확장해서 여러 내부 코드에 한 낱자를 대응시키면, 똑같은 낱자가, 누른 글쇠가 무엇이냐에 따라 서로 다른 조합 과정을 거치게 할 수 있다. 이 규칙을 두면 여러 가지 경우에 유용한데, 예를 들어 세벌식 자판에서 V로 누른 ㄴ는 ㄴ로 조합이 안 되고, /로 누른 ㄴ로만 이중모음이 조합되게 할 수 있으며, 한글 로마자 입력기에서 Y로 입력한 l와 l로 입력한 l의 조합 규칙도 서로 다르게 할 수 있다.

3. 낱자 입력 순서

낱자 입력 순서란 키보드에서 한글 낱자가 하나씩 들어올 때 입력기가 한글을 조합하고 음절을 구분하는 기준을 결정하는 규칙을 말한다. 규칙은 한 개 이상의 키(key)로 구성되며 키는 하나 이상의 1에서 3까지의 숫자로 이루어진다. 여기서 1, 2, 3은 각각 초성, 중성, 종성글자를 뜻한다.

한글은 보통 초성 최대 두 타(된소리일 때 최대), 중성 최대 두 타(겹모음일 때 최대), 종성 최대 두 타(접받침일 때)를 순서대로 눌러 입력한다. 이것을 일반성 있게 표현한 입력 순서 키는 “1 1 2 2 3 3”이다. ‘작두기’라는 단어를 입력하면 키가 ‘1 1 2 3 1 2 1 2’의 순으로 들어온 셈이므로, 이것을 ‘1 1 2 2 3 3’의 일부가 순서대로 여러 번 반복된 것으로 간주하면 1 1 2 3 / 1 2 / 1 2로 음절을 맞게 구분하게 된다.

반면, 이 순서를 “1 1 2 2 3 3” 대신 “1 1 2 2”와 “3 3”으로 나누면 ‘깍두기’ 대신 ‘까ᄃ두기’가 입력된다. 반쯤 풀어쓰기인 셈이다. 1과 2도 따로 갈라지면 ‘ᄃᄃ ᄃᄃᄃᄃᄃᄃ’가 입력되고, 등록된 키가 하나도 없으면 이 입력기는 낱자 결합 규칙과는 무관하게 완전 풀어쓰기를 하여, ‘ᄃᄃ ᄃᄃᄃᄃᄃᄃ’가 그대로 입력되게 된다.

이 순서를 어떻게 규정하느냐에 따라 한글 입력기가 조합해 내는 한글의 물리적인 형태가 위와 같이 달라질 수도 있지만, 같은 모아쓰기 입력이라도 초성, 중성, 종성을 무조건 순서대로 쳐야 모이는 모아쓰기를 만들 수도 있고, 아무 순서대로 쳐도 글자가 모이는 모아쓰기를 만들 수도 있다. “2 1 2 3 3”, “2 2 3 1 3”, “3 1 2 3 2” 등 가능한 순서를 모두 등록하면 되기 때문이다. 또한, 세벌식은 초, 중, 종성 글쇠가 모두 따로 있기 때문에 그렇게 규칙을 정해도 입력기가 아무런 동작상의 문제를 일으키지 않는다. 이 모아쓰기 방식을 세벌식 사용자들은 “모아치기”라고 부른다.

한글이 로마자에 비해 무조건 자음과 모음이 따라야 해서 형태적으로 불리해 보이거나, 세벌식이 글쇠 수가 너무 많은 것이 단점으로 보인다면, 그 단점은 이 모아치기와 같은 새로운 아이디어로 극복해야 한다. 더구나 모아치기는 본 연구가 제안하는 새로운 한글 입력 기능들 대부분의 기반이 되는 매우 중요한 개념이다.

4. 각종 기본 옵션들

Bksp 글쇠를 눌렀을 때 낱자 단위로 지울 것인가, 아니면 음절 단위로 지울 것인가? 조합 중인 글자가 초성과 중성이 갖춰지지 않아 미완성일 때는 들어오는 낱자를 어떻게 할 것인가? 등 “한글 입력기의 기능”이라 할 수 있는 여러 옵션들이 부가적으로 있을 수 있다. 이런 옵션들은 어떤 한글 입력 방식이나 특정 글쇠배열에 종속되지 않는 보편적인 것이어야 함을 원칙으로 한다. 물론 이 연구가 세벌식을 중심으로 한 것인 만큼, 초성·중성·종성이 모두 갖춰진 글쇠배열에서만 쓸 수 있는 옵션은 있을 수 있다.

<날개셋> 한글 입력기의 기능

앞에서 살펴본 한글 입력의 네 요소를 사용자가 마음대로 정의할 수 있도록 본 연구에서는 <날개셋> 한글 입력기를 제안하였다.

한편, <날개셋> 한글 입력기가 새롭게 지원하는 기능들은 조합 중인 한글이든 조합이 끝난 한글이든 글자와 낱자 단위로 자유자재로 바로 지우고 고치고 입력할 수 있게 하는데 목적이 있다. 이 일을 가능하게 하는 중점에는 앞에

서 소개한 “모아치기”라는, 어떤 순서대로 낱자를 입력하는 것도 허용해 주는 입력 순서 규칙이 있다.

이들 기능들은 옵션이나 특수 글쇠의 형태로 제공되며, 지우기, 입력하기, 고치기, 커서 이동이라는 네 부류로 나뉜다.

1. 지우기

1) 낱자와 글자 단위로 자유롭게 지우기

한글 입력 방식을 설계할 때 매우 중요한 원칙이 있다. 모아쓰는 입력 방식이 풀어쓰는 것보다 오히려 불편해서는 안 된다는 것이다. 그러나 지금 우리가 컴퓨터에서 쓰는 한글 입력기는 그렇지 못하다. 한글을 낱자나 글자 단위로 자유롭게 지우지 못해서, 틀린 낱자 하나를 고치기 위해 글자 전체를 지워야 하기 때문이다.

특히 한글을 빨리 치다 보면, 한 낱자에서 오타가 발생해도 손은 그 오타를 바로 지우기보다는 조합하고 있던 틀린 글자를 완성하고 나서야 멈추기 쉬워 이러한 기능이 더욱 필요하다. 따라서 <날개셋> 입력기에서는 낱자 단위로 지우는 것(1), 글자 단위로 지우는 것(2), 조합 중인 글자를 포함한 어절을 한꺼번에 지우는 것(3)이 모두 가능하게 했다.

2) 앞 글자에 달라붙기

Bksp키를 누른 결과, 커서가 한글의 바로 뒤에 있게 되면 입력기가 자동으로 그 앞 글자를 조합하던 상태를 만들어 주는 옵션이다.

4는 뒷글자 ‘기’를 모두 지우자 자동으로 입력기가 앞글자 ‘아’를 조합하는 상태로 만들어 줌으로써 받침 ㄴ을 손쉽게 추가한 모습이다. 5번처럼 한글을 지우지 않더라도 *이 지워진 뒤에 앞 글자가 한글이면 이 기능이 적용되어야 한다.

그뿐만 아니라 현재 글자를 조합 중인 상태가 아니더라도 바로 앞 글자가 한글이면 이 글자의 마지막 낱자만 지운 뒤, 그 글자를 조합하는 상태가 재현되어야 할 것이다. 6과 7은 그 예를 잘 보여준다. 받침 ㄴ을 한 번에 입력할 수 있는 세벌식에서는 받침이 바로 지워지고, 그럴 수 없는 두벌식에서는 ㅎ만 지워져 있는 것을 알 수 있다.

그런데 이 기능을 두벌식 체계에서 적용하면 매우 부자

대한민국
1. 대한민국
2. 대한민
3.

Fig. 1. 다양한 단위로 지우기.

4. 아기 → 아ᄃ → 아 → 안
5. 안* → 안
6. 앞 → 아
7. 앞 → 안

Fig. 2. 앞 글자에 달라붙기.

8. 일기 → 일
9. 떡보끼 → 떡북
10. 고빠 → 고ㅍ

Fig. 3. 두벌식에서의 부자연스러운 달라붙기.

연스러운 일이 일어난다.

8에서 '일기'를 치고 있다가 Bksp를 눌렀는데 갑자기 기가 사라지고 앞 글자의 '일'의 받침으로 결합해 버리는 것을 알 수 있다. 9의 기도 마찬가지이다. 도깨비불 현상이 역으로 일어나는 셈이다. 더구나 이 일을 하기 위해서는 현재 자음이 앞 글자의 받침으로 갈 수 있는지도 입력기가 판단해야 한다. 10에 나와 있는 빠처럼, 받침으로 쓰이지 않는 낱자도 있기 때문이다. 이렇듯 두벌식 체계는 새로운 기능을 구현하는데 한계를 드러내고 있다.

2) 특정 낱자 바로 지우기

한글 낱자를 지우는 데 Bksp키를 써서 꼭 최근에 입력된 것부터, 다시 말해 중성-중성-초성 순서대로만 지워야 할 필요가 있을까? 특수 글쇠 세 개를 글자판에 배당하면 입력된 순서에 관계없이 지정한 성분의 낱자를 바로 지울 수 있다.

이 글쇠를 쓰면, 조합 중인 '갑'을 '압'으로 바꾸기 위해 애써 입력한 '갑'을 몽땅 지울 필요가 없다. 오른손을 Bksp 키까지 옮길 필요도 없다. 초성을 지우는 키를 눌러 'ㅏ'나 'ㅑ'를 만든 뒤 'ㅇ'만 다시 누르면 된다. 비단 조합 중인 글자뿐만 아니라 완성된 '갑' 앞에 커서를 두고 초성 삭제 글쇠를 눌러도 'ㅏ'나 'ㅑ'를 조합하는 상태가 되고, 초성과 중성까지 삭제하면 글자가 완전히 지워지게 된다. 조합 중이 아닌 완성된 글자의 낱자를 이렇게 지우려면 편집기의 입력 상태가 '삽입'이 아닌 '겹침'이어야 한다.

이런 간단하고도 편리한 기능이 이제까지 컴퓨터에서 구현될 수 없었던 이유는 'ㅏ'나 'ㅑ'와 같은 미완성 글자를, 보다는피 완성형 한글 코드에서 표현하기가 불가능했기 때문이다. 또 다른 이유가 있다면 두벌식 자판 때문이라고도 할 수 있다. 글자를 입력하는 글쇠 자체가 자음과 모음밖에 없는데 한글 입력기를 만드는 사람이 초성-중성-중성을 세 벌로 따로 지우는 특수 글쇠를 넣는다는 생각 자체가 어울리지 않는 것이다.

또다른 이유로는, 초성을 지운 'ㅏ'나 'ㅑ'에서 '압'을 다시 입력할 수 있는 근거가 바로 앞에서 소개한 "모아치기"이기 때문이다. 모아치기는 두벌식에는 적용할 수 없다. '고'나 '아'를 치면 '꽁'이 되어 버리기 때문에 두벌식은 무조건 초/중/중성을 순서대로 치며 입력자에게 그 순서로 음절 구분에 대한 정보를 제공해야 한다. 결국 낱자를 바로 지

우는 글쇠 역시 자연스럽게 일관성 있게 구현하기가 불가능해지는 것이다.

2. 입력

1) 미완성 한글 입력 방지

글자를 입력하다가 글쇠를 하나 더 눌러버려 한 글자가 미완성 글자 여러 개로 분해되는 오타는 자주 낸 적이 있을 것이다. 예를 들어 '간'을 치려다가 잘못해서 '가ㅏㅑ'와 같은 글자가 생기는 상황을 들 수 있다. 앞에서 소개한 우수한 기능만으로도 이 오타를 신속하게 고칠 수 있지만, 이런 오타는 형태적인 것이기 때문에 입력기 차원에서 근본적으로 차단하는 방법이 있다. 조합 중인 글자가 최소한 초성과 중성이 갖춰지기 전에는 글쇠가 더 들어와도 현재 글자 조합을 끝내지 않거나, 한글인 경우 지금 입력 중인 글자를 고치게 하는 것이다.

조합 상태를 탈출하려면 화살표 키로 커서를 움직여야 한다. 따라서 문장 말미에서 자주 내는 'ㄷ·ㅏ'와 같은 오타도 예방된다.

2) 무한 낱자 수정

이 기능에서 좀더 나아가면, 아예 초성을 제외한 모든 낱자는 입력 중인 글자를 고치는데 쓰이도록 할 수도 있다. 이것을 본 논문에서는 "무한 낱자 수정"이라고 일컫기로 한다. 이 기능을 고치기가 아니라 입력 기능으로 분류한 이유는, 이 기능 자체가 고치는 기능이라기보다는, 글쇠 입력을 무한히 받음으로써 고치는 효과를 내는 것에 가깝기 때문이다. 세벌식에서 순서를 뒤바꿔 치는 오타는 왼손, 즉 중성과 중성에서 자주 일어나지 글자가 시작되는 초성과 중성 사이에서는 잘 일어나지 않기 때문에 이렇게 활용하는 것이 충분히 고려할 가치가 있다.

이 기능을 활용하면 멀리 떨어져 있는 Bksp를 거의 누르지 않고 오타를 바로 고치는 것이 가능해진다. '밥'을 잘못해서 '땡'으로 입력하더라도 ㅑ는 바로 ㅏ로 고치면 되고, 초성 ㅌ만 앞에서 소개한 초성 삭제 글쇠를 눌러 자리를 비운 뒤, ㅑ를 누르면 된다. 두벌식에서는 상상조차 할 수 없는 이 기능 뒤에 숨어 있는 것은 세벌식 모아치기라는 탄탄한 기반이다.

2. 고치기

1) 글자와 낱자 어떤 단위로나 수정

글자나 낱자 단위로 자유롭게 지울 수 있다면 수정도 자

유롭게 가능해야 할 것이다. 이것을 구현하는 가장 좋은 방법은 편집기 차원에서 '글자 겹침', '날자 겹침'을 지원해서 입력기가 거기에 맞게 동작하는 것이다. 평소에는 우리가 쓰던 대로 '글자 삽입' 또는 '글자 겹침' 모드를 쓰지만 이미 완성된 글을 검토한다거나 할 때, '날자 겹침'이 유용할 것이다. '간'을 '단'으로 바꾸는 데는 당연히 한 타밖에 들지 않아야 한다.

이렇게 '날자 겹침' 모드에서 한글을 고치면 입력기는 평소의 조합 상태와는 약간 다른 '수정 모드'로 진입한다. 수정 모드일 때의 커서는 표준 조합 모드일 때와는 모양이 약간 다르다. 또한 모아치기나 무한 날자 수정처럼, 모아쓰기 전용 옵션들이 해제된다. 고치는 것은 필요한 날자 몇 개만 입력하는 것이기 때문이다.

날자 단위로 고치는 기능은 두벌식에서는 물론 제대로 사용할 수 없다. 초성과 종성의 구분이 없기 때문에 고치고 싶은 성분을 직접 지정해 주는 것과 같은 복잡한 처리를 해야 한다.

2) 앞뒤 글자로 날자 옮기기

날자의 입력 순서가 뒤바뀐 오타는 앞에서 말한 것처럼 모아치기를 통해 저절로 보완이 될 수 있다. 하지만 앞뒤 글자의 날자가 뒤바뀐 건 이제까지 달리 바로잡을 방법이 없었다. 특히 받침에서 '가을'을 '갈으'와 같이 치는 경우는 세벌식에서는 의외로 자주 발생하는 편이다. 한글 입력기 차원에서 이 오타를 바로잡는 길을 열어 줘야 하는 이유는, 풀어쓰기에서 손쉽게 가능한 것은 모아쓰기에서도 손쉽게 가능해야 하기 때문이다.

풀어쓰기에서는 앞뒤 글자가 뒤바뀐 오류 정도는 복사해서 붙이는 매크로 기능으로 바로잡을 수 있다. 이 기능은 한글 입력기에서는 특수 글쇠로 구현하는 된다.

다음은 '여름'에서 '름'을 조합하다가 종성을 왼쪽으로 옮기는 특수 키를 눌렀을 때, 이 키가 동작하는 방식을 나타낸 것이다.

똑같은 상황이라도 현재 편집기가 삽입 상태이나(1) 겹침 상태이나, 겹침 상태라면 날자 단위로 겹치느냐(2) 글자 단위로 겹치느냐(3)에 따라 동작 방식이 약간 달라진다. 받침 ㅁ을 조합하고 있는 상태에서 글자를 완성할 수 있으려면 모아치기가 지원되어야 할 것이다.

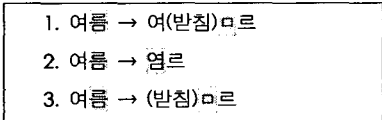


Fig. 4. 받침을 앞으로 옮기는 특수 키.

3. 커서 이동

1) 앞뒤 글자를 조합하는 상태로

이것 역시 특수 글쇠 한 쌍의 형태로 구현된다. 화살표 키를 누른 것처럼 커서가 왼쪽 또는 오른쪽으로 한 글자 단위로 이동하는데, 한글의 위치에 있게 되면 그 글자를 조합하는 상태를 자동으로 만들어 준다. 그래서 이제까지 설명했던 여러 가지 편리한 고치기 기능으로 그 글자를 빨리 고칠 수 있다.

다음은, 깜빡 잊고 앞 글자의 종성을 입력하지 않은 채로 다음 글자로 넘어가 버렸을 때, 현재 입력 중인 글자를 지우지 않고, 화살표 키도 쓰지 않고 앞글자를 재빨리 고치고 다시 원래 글자로 돌아온 모습을 나타낸 것이다. 특수 키를 쓰는데 익숙해지면 '자'를 지워서 앞 글자로 가는 것보다 이렇게 고치는 게 속도가 더 빨라지게 된다.

활용 가능성

우리는 이와 같은 아이디어를 모두 구현한 <날개셋> 한글 입력기를 C++ 라이브러리의 형태로 개발하고, 이를 바탕으로 메모장 수준의 간단한 에디터와, 세벌식 타자 연습 프로그램을 만들어 저자의 홈페이지를 통해 무료로 배포하고 있다. 국내외의 많은 세벌식 사용자들이 기능들을 만족하며 쓰고 있으며, 세벌식이 오토마타 없이 한글을 입력할 수 있고, 오토마타가 동원되면 컴퓨터 시대에 훨씬 뛰어난 성능을 발휘할 수 있다는 사실에 감탄했다고 우리에게 알려 왔다.

이제 우리는 이 입력기를 윈도우/오피스 XP의 고급 텍스트 서비스 환경(TSF)으로 어느 정도 이식했으며, 이 논문 역시 이 입력기로 수월하게 입력했다. 기존 IME를 텍스트 서비스로 조만간 대체하는 것이 MS 사의 정책인 만큼, <날개셋> 입력기를 쓸 수 있는 환경도 점차 넓어질 거라는 기대를 할 수 있다. 워드 프로세서나 웹 에디터를 개발하는 국내 개발자들도 추후 버전부터는 TSF를 지원해

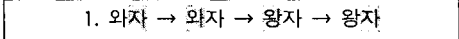


Fig. 5. 앞 글자를 조합하는 글쇠를 사용한 예.

Table 1. 기본 특수 글쇠의 위치와 기능

위치	기능 설명
Shift+T	초성 삭제
Shift+4, 5	중성, 종성 삭제
Shift+7, 8	받침을 앞/뒤 글자로 옮김
Shift+<, >	앞/뒤 글자를 조합하는 상태

야 할 것이다.

앞에서 설명한 특수 기능들을 구현하기 위해 <날개셋>에서 기본으로 지정한 특수 글쇠의 위치와 기능은 Table 1과 같다. 세벌식 최종 글쇠배열에서 잘 쓰이지 않는 자리에 이들을 배당했다.

결론과 제언

본 연구 결과가 반영된 <날개셋> 입력기는 기존 입력기보다 훨씬 한글의 성능을 잘 살리면서 한글을 입력하게 함으로써 타자 능력을 향상시키고, 사용자의 타자 작업 또는 연습을 훨씬 즐겁게 해 줄 것이다. 이미 이들 기능은 경험적으로는 검증¹²⁾을 받았으므로 이제 가장 필요한 것은 이론적인 검증이다.

본 연구 결과를 살펴보면 한글 입력 자체뿐만 아니라 고치고 지우는 기능에도 비중이 매우 부여돼 있음을 알 수 있다. 이것들이 모두 조합과 관련돼 있기 때문이다. 한글 입력기는 조합과 관련된 기능이 생명이다. 여기에 대한 능력이 더욱 체계적으로 검토되어야 하며, 본 연구 결과가 타자 능력을 객관적으로 수치상으로 얼마나 올려주는지도 검증되어야 할 것이다.

한편, 나라의 표준 정책을 주관하는 당국에 대해서는 하루빨리 세벌식 자판을 단독 표준이든 두벌식과 복수 표준이든 나라를 대표하는 표준으로 삼아 줄 것을 제언한다. 세벌식 자판이 구조나 배열 면에서 우수함은 수십 년 동안 검증된 사실이며, 세벌식/두벌식 문제는 단순한 속도 문제

차원을 넘어 한글 기계화의 백년대계를 좌우하는 중대한 문제이다.

세벌식 체계에서 두벌식을 생각해 내는 것은 쉽지만, 두벌식 체계에서 세벌식을 생각해 내는 것은 매우 어렵다. 사고의 범위가 제한되는 것이다. 세벌식이 없었다면 모아치기에서부터 무한 날자 수정으로 이르는 입력 옵션이라든가 날자 단위 삭제와 같은 기능을 구현하는 데 큰 모순이 생기고, 심지어는 <날개셋> 입력기 자체가 개발되지 못했을지도 모른다.

세벌식의 존재를 지금이라도 나라에서 정식으로 인정해 주는 것은 진리를 아는 후세에 의해 역사에 길이 남는 업적으로 평가될 것이며, 현재 정부에 대해 쌓인 국내외 세벌식 사용자들의 불신을 해소해 줌과 동시에 <날개셋> 한글 입력기의 개발과 보급에도 매우 고무적이고 긍정적인 일이 될 것이다.

REFERENCES

- 1) 한소프트(1994) : "새로운 한글 자판(한 자판)의 제안"
- 2) (<http://www.hansoft.com/doc/etc/hkb.html>)
- 3) 신광조(1995) : "신 세벌식"
- 4) (<http://my.netian.com/~sdjun/shin/shin.html>)
- 5) 조종진(2000) : "복벌식 한글 입력기"
- 6) (<http://my.netian.com/~sdjun/act/act.html>)
- 7) 안마태(2003) : "안마태 한글 소리 글판"
- 8) (<http://ai.kaist.ac.kr/ahnmatae/>)
- 9) 송 현(1989) : "한글 기계로 옮겨 쓰기", 대원사
- 10) 김용목(2001) : "제2의 한글 프로그래밍을 꿈꾸며"
- 11) (<http://moogi.new21.org/story9.htm>)
- 12) 김용목(2002) : "공 병우 식 한글 기계화의 철학"
- 13) (<http://moogi.new21.org/story15.htm>)