순환구조의 은닉 마르코프 모델을 이용한 온라인 영어 필기인식

오 세창○ 하 진영 김 진형
한국과학기술원 전산학과

Unconstrained Handwriting Word Recognition with Interconnected Hidden Markov Models

S.C. Oh J.Y. Ha J.H. Kim
Computer Science Dept. of KAIST

요 약

본 논문에서는 전자펜으로 제한없이 쓴 영어 단어를 인식하기 위한 방법을 소개한다. 온라인 필기인식을 위해 사용된 기본적인 방법론은 변형 흡수 능력이 뛰어난 은닉 마르코프모델 (HMM) 이다. 본 논문에서는 문자 뿐만 아니라 문자 사이를 연결해주는 연결획 부분에 대해서도 모델링함으로써 연결해서 쓴 단어의 인식을 가능하게 하였고, 이들을 순환구조로 연결하여 단어의 길이에 관계없이 인식을 할 수 있도록 하였다.

단어를 인식하기 위해서 사용된 방법은 Viterbi 알고리즘이다. 이 알고리즘을 사용하여 순환구조로 연결된 HMM 상에서 최적 경로를 찾는다. 인식결과는 최적 경로 상에 어떤 문자모델과 연결획 모델들이 놓여있는가를 봄으로써 얻을 수 있다. 또한 단어 사전을 경로 탐색의 각 단계에서 참조하여 불필요한 후보를 미리 제거함으로써 보다 정확한 인식 결과를 얻게 된다.

1. 서론

영어 단어를 필기할 때 사람들은 일반적으로 정자체와 흘림체 등 여러가지 필기 스타일을 혼용한다. 특히 흘림체의 경우 두개 이상의 문자가 서로 연결될 수 있다. 또한 온라인 문자 인식의 기본 가정인 획순의 제한을 깨는 지연획(delayed stroke)도 사용된다. 마우스나 키보드의 역할을 펜으로 대체하는 노트패드 컴퓨터가 실용화되기 위해서는 이렇게 자유롭게 필기한 글씨를 인식하기 위한 기술이 필요하다.

그러나 이를 인식하는 문제가 어렵기 때문에 대부분의 영어 필기 인식에 관한 연구가 필기 방법에 제한이 많은 boxed 스타일이나 run-on 스타일에 집중되어 왔으며, 제한없는 필기의 인식에 관한 연구는 대학이나 연구소에서 부분적 으로 진행되어왔다 [Cam91, Sch91, Hig85, Tap82].

제한없는 필기를 인식하기 위해서는 다음의 몇가지 기본적인 문제를 해결해야 한다. 첫째, 한 단어내에서 각 문자의 경계를 알아내는 문자 분리 문제이다. 이 문제는 인식과정의 도움없이 정확한 해답을 얻을 수 없다. 또 하나의 문제는 다양한 필체와 변형을 어떻게 흡수할 것인가 하는 문제이다. 세번째는 문자들을 연결해서 쓸때 발생하는 연결획 부분의 처리 문제이다. 이 밖에 지연획의 처리, 필기 입력의 크기 정규화, 단어의 전체적인 기울기(orientation)와 획의 기울기(slant)의 교정 등 결코 간단하지 않은 문제들이 있다 [Bro88].

이러한 문제들을 해결하기 위해서 사용된 기본적인 방법론은 HMM이다. HMM은 음성인식 분야에서 가장 널리 그리고 성공적으로 사용되고 있는 방법론이다 [Bah83]. 그리고 80년대 부터 문자인식 분야에 적용되어왔다 [Rab90, Kun88, Nag86]. 이렇게 HMM이 널리 쓰이는 이유는 변형이 심한 경우에도 훌륭한 모델링 성능을 발휘하며, Viterbi 알고리즘과 같은 효과적인 디코딩 알고리즘이 있기 때문이다.

본 논문에서는 문자뿐만 아니라 연결획 부분에 대해서도 HMM에 의해 모델링을 함으로써 연결획 부분을 자연스럽게 흡수하도록 하였다. 또한 필기 단어를 문자와 연결획 부분이 교대로 존재하는 시퀀스라고 볼 수 있으므로 문자 모델과 연결획 모델을 순환구조로 연결하여 단어 생성 모델을 만들었다. 이 단어 생성 모델에 Viterbi 알고리즘을 적용하여 인식과 문자 분리 과정을 동시에 수행한다. 지연획은 온라인의 기본 가정에 위배되므로 제거한 후 Viterbi 알고리즘에 의해 최적 경로 탐색을 할 때 이에 대한 정보를 참조하는 방법을 사용했다. 또한 보다 정확하고 효과적으로 단어인식 결과를 얻기위해 문맥 정보를 후처리 단계에 사용하지 않고 경로 탐색의 각 단계에서 사용한다.

2. 문제의 정의

필기한 단어는 다음과 같이 문자와 연결획 부분이 교대로 존재하는 시퀀스로 볼 수 있다.

$$Handwriting := Character \cdot \{Ligature \cdot Character\}^* \quad (1)$$

여기에서 Handwriting은 단어에 대한 필기 데이타이며, 연결획은 두 문자 사이에 펜을 들고 이동한 것과 펜을 들지 않고 이동한 것 모두를 포함한다.

이때 W를 k개의 문자로 이루어진 필기 데이타를 표현하기 위한 모델들의 시퀀스라고 하면, W는 다음과 같이 문자 모델과 연결획 모델의 시퀀스로 표현된다.

$$W = C_1 L_1 C_2 \cdots L_{K-1} C_K \quad (2)$$

여기에서 C_i는 문자 모델이고 L_i는 연결획 모델이다. 그림 1은 각 문자 부분이 문자 모델 C_1, C_2, C_3, C_4, C_5에 의해 표현되고, 이들 사이에 존재하는 각 연결획 부분이 연결획 모델 L_1, L_2, L_3, L_4에 의해 표현되는 것을 보여준다. 이 그림에서 L_4와 같이 펜을 들고 이동한 부분도 연결획에 해당된다.

필기 입력 X를 인식하는 문제는 가능한 모든 모델 시퀀스 W중 다음과 같은 조건확률 $Pr(W|X)$를 극대화하는 최적 모델 시퀀스

그림 1: 단어 필기의 예

$\hat{W}$을 구함으로써 풀 수 있다.

$$Pr(\hat{W}|X) = \max_{W} \frac{Pr(X|W)Pr(W)}{Pr(X)} \quad (3)$$

여기에서 $Pr(W|X)$는 정합된 모델들과 필기 입력과의 모양 유사도를 구함으로써 vspace3mm 얻을 수 있고, $Pr(W)$는 그 단어가 주어진 문맥상 얼마나 자주 나오는지에 근거한 언어 모델에 의해서 얻을 수 있다. 식 (3)에서 $Pr(X)$는 W에 독립이므로 이 식은 다음과 같이 정리된다.

$$\begin{aligned} Pr(\hat{W}|X) &\propto Pr(X|\hat{W})Pr(\hat{W}) \\ &= \max_{W}[Pr(X|W)Pr(W)] \\ &= \max_{C,L} \max_{\tau} [Pr(X_{1,t_1}X_{t_1+1,t_2}\cdots \\ &\quad X_{t_{2K(\tau)-2}+1,T_{hw}}|C_1L_1\cdots C_{K(\tau)})\cdot \\ &\quad Pr(C_1L_1\cdots C_{K(\tau)})] \end{aligned} \quad (4)$$

여기에서 τ는 필기 입력 X에 대한 분할을 표시한다. 결국 필기 인식 문제는 주어진 필기입력에 대한 모든 가능한 분할 방법 중 최적의 모델 시퀀스 $\hat{W}$를 찾는 문제라고 할 수 있다. 일반적으로 가능한 분할 방법은 모두 계산할 수 없을 정도로 많은데, 이것은 필기 데이타를 구성하는 모든 점들이 문자의 경계가 될 수 있기 때문이다.

3. 단어 생성 모델

현실적인 시스템을 만들기 위해서 본 논문에서는 그림 2와 같이 연결구조가 간단한 단어 생성 모델을 제안하였다. 언어 모델을 고려하여 단어 생성 모델을 만들 수 있으나 이것은 모델의 복잡도를 가중시킨다. 따라서 본 논문에서는 언어 모델과 독립적인 단어 생성 모델을 사용하였고 언어 모델은 인식 과정에서 따로 고려한다.

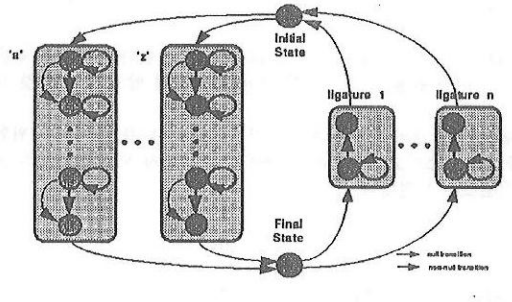

그림 2: 단어 생성 모델

단어 생성 모델의 구성은 각 문자 모델과 연결획 모델 그리고 이들을 순환구조로 연결하기 위해 전체 모델의 시작 노드와 종료 노드에 해당하는 두개의 grammar node를 사용하였다. 또한 grammar node와 각 문자, 연결획 모델 간의 연결에는 전이 과정에서 심볼을 생성하지 않는 null transition을 사용하였다. 단어 생성 모델의 구성요소인 각 문자 모델과 연결획 모델은 왼쪽에서 오른쪽으로 전이되는 형태의 HMM이다.

이 모델에서 단어인식은 시작 노드에서 시작해서 문자 모델과 연결획 모델을 교대로 통과한 후 종료 노드에 도달하는 최적 경로를 찾는 과정이다. 이 구조에서 임의의 길이를 갖는 문자열에 해당하는 경로가 가능하다. 그러나 이러한 문자열 중에는 단어가 될 수 없는 경우가 다수 포함된다.

4. 경로 탐색 알고리즘

본 논문에서의 단어 인식을 위한 경로 탐색 방법은 Viterbi 알고리즘을 기반으로 한다. Viterbi 알고리즘은 최적 경로를 찾는 네트워크 탐색 기법인데, 이 알고리즘은 기본적으로 하나의 최적 경로를 찾는다. 이 알고리즘을 그대로 적용하여 단어 인식을 할 경우 언어 모델을 적용하여 올바른 단어가 될 수 없는 경로를 제거하면 모든 경로가 다 제거되고 인식 결과를 낼 수 없는 경우가 발생할 수 있다. 이때 경로 후보를 하나만 구하지 않고 여러개를 구하면 중간에 몇개의 후보가 제거되더라도 그 다음 후보가 살아남아 인식 결과를 얻을 수 있다. 따라서 본 논문에서는 다음과 같이 C개의 최적 경로를 구하도록 Viterbi 알고리즘을 수정하였다.

먼저 i번째 노드에서 시간 t에 구해진 c번째 최적 경로를 $\omega_i^c(i)$라고 하고 그 확률을 $\delta_t^c(i)$라고 하면 $\delta_t^c(i)$는 다음과 같이 정의된다.

$$\delta_t^c(i) = \max^{c}_{q_1,q_2,\cdots,q_{t'-1}} Pr(q_1,q_2,\cdots,q_{t'}=i,O_1,O_2,\cdots,O_t|\lambda) \quad (5)$$

여기에서 $\max^c$는 c번째 최대치를 구하는 함수이다.

1. 초기화:

$$\delta_0^1(1) = 1 \quad (6)$$
$$\delta_0^c(i) = 0, \quad \forall i \text{ and } c, \text{ except } i = c = 1. \quad (7)$$
$$\omega_0^c(i) = null, \quad \forall i \text{ and } c. \quad (8)$$

2. 재귀 단계:

$$\delta_t^c(j) = max^{c}_{i\in S_T(j),k\in S_N(j),n=1,2,..,C}[\delta_{t-1}^n(i)a_{ij}b_{ij}(O_t), \delta_t^n(k)a_{kj}], \quad 1 \le j \le N, 1 \le t \le T, \quad (9)$$

여기에서 $S_T(j)$와 $S_N(j)$는 노드 S_j에 각각 null arc와 non-null arc로 연결된 이전 노드들의 집합이다.

$$\omega_t^c(j) = append(\omega_{t'}^{c'}(i'), j), \quad (10)$$

여기에서 i', t', c'는 각각 노드 S_j의 이전 노드의 번호, 시간, 후보순위를 나타낸다.

3. 종료:

$$P^c = \delta_T^c(N), Q^c = \omega_T^c(N), \quad (11)$$

여기에서 N는 전체 노드수를 나타낸다.

이 알고리즘에의해 구한 최적 경로중 시간 T에 단어 생성 모델의 종료 노드에서 구한 경로 후보가 주어진 입력에 대한 인식 결과이다.

5. 필기 인식 시스템의 구현

5.1 시스템의 구성

영어 필기인식 시스템은 그림 3과 같이 구성된다. 영어 단어에 대한 필기입력은 전처리 과정을 거친 다음 16 방향 코드로 부호화 된다.

이 방향 코드의 시퀀스를 입력으로 단어 인식기는 단어 생성 모델에 표현된 모양에 관한 지식과 단어 사전에 포함된 문맥 지식을 사용하여 단어 후보를 생성한다.

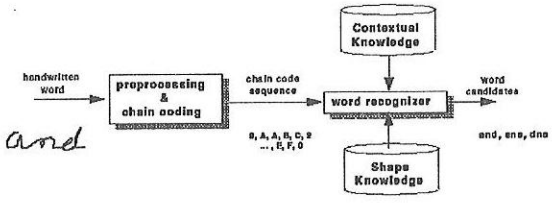

그림 3: 필기 인식 시스템의 구성

5.2 필기입력의 전처리 및 부호화

전처리 단계는 입력 데이타에서 노이즈를 제거하고 가급적 표준 형태로 만들기 위해 필요하다. 본 논문에서 사용된 전처리 방법은 단어 기울기 교정, 크기 정규화, 혹과 같은 노이즈의 제거, 평활화, 등간격 재샘플링 그리고 지연획의 제거 등이다.

전처리가 끝난 다음 모든 획에 대해 방향 코드의 시퀀스를 생성한다. 이 과정에서 실제획에 대해서 16 방향 코드를 부여하고 펜을 들고 이동한 가상획에 대해서 따로 16 방향 코드를 부여한다. 그림 4은 코드 체계와 t자를 필기했을 때의 부호화 예를 보여준다.

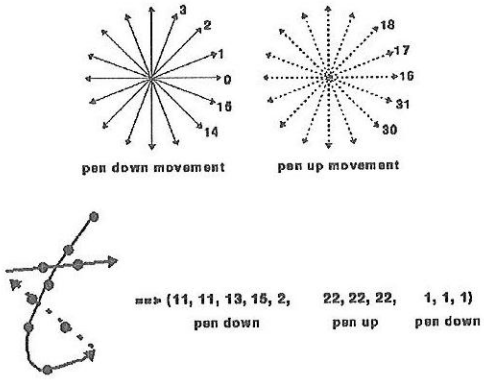

그림 4: 방향코드와 부호화의 예

5.3 단어 인식

단어인식은 시작 노드에서 시작해서 부호화된 입력 시퀀스를 모두 소비하면서 종료 노드에 도달하는 최적 경로를 찾는 과정이다. 방향코드가 하나씩 단어 생성 모델에 의해 소비될 때마다 Viterbi 알고리즘은 가능한 상태전이를 계산하고 이중 가장 확률이 높은 경로 후보 n개씩을 각 state에서 유지한다. 이때 각 grammar node에서 유지하고 있는 경로 후보들로부터 단어의 앞부분에 해당하는 문자열을 얻을 수 있는데, 이들 중 올바른 단어가 될 수 없는 것들이 존재한다. 이들을 단어 사전을 참조하여 미리 제거함으로써 불필요한 계산을 줄이고 보다 정확한 인식 결과를 얻게된다.

5.4 지연획의 처리

지연획은 획의 유형 그리고 바로 이전획과의 위치상의 관계를 보고 판별할 수 있다. 즉, ij-dot와 tx-bar 형태의 획이 바로 이전 획의 끝점보다 어느정도 이상 왼쪽에 위치하면 지연획이라고 판단할 수 있다. 또한 이러한 유형의 획들이 연속해서 나오면 두번째 이후의 획들은 모두 지연획이다. 그림 5의 예에서 두개의 bar는 위에서 설명한 논리에 의하여 지연획으로 판별된다.

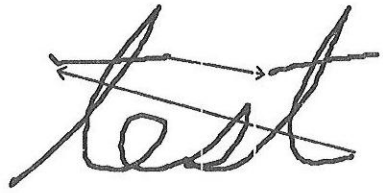

그림 5: 지연획의 예

식별된 지연획은 유형과 위치정보를 얻은 후 제거된다. 이렇게 dot나 bar 형태의 획이 없는 필기 입력에 대해 정합을 하기 위해서 dot나 bar가 없는 i, j, t, x의 문자모델도 사용되었다.

Viterbi 알고리즘에 의해 생성되는 각 경로 후보에 대해 최근에 지나온 문자 모델이 dot나 bar가 없는 i, j, t, x 모델일 경우에는 그에 해당하는 지연획이 그 위치에 있었는지를 검사한다. 만약 해당하는 지연획이 없다면 그 모델을 최종적으로 거쳐나온 경로 후보는 제거된다.

6. 실험 결과

문자 모델과 연결획 모델의 훈련에는 각각 11,002개의 문자 샘플과 4505개의 연결획 샘플이 사용되었다. 이 데이타는 8명의 필기자가 쓴 단어로 부터 수작업에 의한 문자 분리 과정을 거쳐서 얻었다. 인식률을 구하기 위한 실험 데이타는 다른 3명의 필기자로부터 얻은 문자 샘플과 단어샘플 그리고 단어샘플을 수작업에 의해 분리한 결과를 사용했다.

6.1 문자 인식 실험

문자인식 실험에서는 1,556개의 문자 샘플을 사용하였다. 실험에서 사용한 모델은 26개의 문자 모델과 지연획을 처리하기 위한 4개의 추가 문자 모델이다. 이들은 모두 노드 수가 10개이며 왼쪽에서 오른쪽으로 전이되는 구조를 갖는 HMM이다. 표 1은 5번째 후보까지의 누적 인식률을 보여준다.

표 1: 문자 인식률(%)

candidate	rate
1st	87.66
2nd	95.37
3rd	97.75
4th	98.71
5th	99.16

6.2 단어 인식 실험

단어인식 실험에서는 400여개의 단어 샘플을 사용하였다. 여기에서 사용된 단어 생성 모델은 문자 인식 실험에서 사용했던 29개의 문자 모델과 노드 수가 2개인 하나의 연결획 모델로 구성된다. 단어 사전은 38,000 단어가 사용되었다. 인식을 위한 경로 탐색 과정에서 단어 생성 모델의 각 노드에는 5개씩의 경로 후보가 유지된다. 표 2은 5번째 후보까지의 누적 인식률을 보여준다.

표 2: 단어 인식률(%)

candidate	rate
1st	81.22
2nd	85.64
3rd	86.46
4th	86.74
5th	86.74

이 실험에서 하드웨어는 PC 486을 사용하였고 Windows 환경에서 단어당 0.74초의 인식 속도를 얻었다. 여기에서 사용한 실험 데이타의 평균 단어길이는 4.1이다.

7. 결론

본 논문에서는 제한없이 필기한 영어 단어를 인식하기 위한 현실적인 방법을 제안하였다. 이 방법은 변형 흡수 능력이 뛰어난 HMM을 기반으로 한다. 이 방법의 핵심인 단어 생성 모델은 문자 모델과 연결획 모델을 순환구조로 연결함으로써 간단한 구조로 임의의 길이를 갖는 단어를 인식할 수 있도록 구성되었다.

실험 결과를 보면 인식속도는 현실적인 범위 안에 들어오지만 인식률은 아직 많은 개선이 요구된다. 인식률을 높이기 위한 방법으로는 모델 관리 기법, 부호화 방법의 개선, 전체적인 특징을 이용하여 단어사전을 줄이는 방법 등을 생각할 수 있다. 먼저 모델 관리 기법은 훈련 데이타 중 모양이 많이 다른 데이타들을 분류하여 각각 다른 모델을 만들어줌으로써 각 모델이 보다 정확한 모델링을 할 수 있도록 한다. 두번째로 부호화 방법의 개선은 방향코드 뿐만 아니라 보다 표현력이 좋은 특징을 사용하자는 것이다. 세번째는 필기 데이타의 전체적인 특징을 이용하여 단어 사전의 크기를 줄여주는 방법이다. 단어 사전의 크기가 줄어들면 그 범위 안에서 보다 정확한 인식 결과를 얻을 수 있다.

감사의 글

본 연구는 현대전자, 한국컴퓨터, 삼보컴퓨터, 포스데이타, 대우통신, 삼성전자로 구성된 노트패드 컨소시움과 한국과학기술원 인공지능 연구센타를 통한 한국과학재단의 연구비 지원을 받았음을 밝힙니다.

참고문헌

[Bah83] L. R. Bahl, F. Jelinek, and R. L. Mercer, "A maximum likelihood approach to continuous speech recognition," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. PAMI-5, pp. 179–190, Mar. 1983.

[Bro88] E. R. Brocklehurst and P. D. Kenward, "Preprocessing for cursive script recognition," Tech. Rep. DITC 132/88, National Physical Laboratory, Nov. 1988.

[Cam91] J. Camillerapp, G. Lorette, G. Menier, , H. Oulhad, and J. C. Pettier, "Off-line and on-line methods for cursive handwriting recognition," in *Third Int. Workshop on Frontiers in Handwriting Recognition*, (Chateau de Bonas, France), pp. 253–264, 1991.

[Hig85] C. A. Higgins and R. Whitrow, "On-line cursive script recognition," in *Human-Computer Interaction ~ INTERACT'84* (B. Shackel, ed.), pp. 139–143, Elsevier Science Publishers, 1985.

[Kun88] A. Kundu and P. Bahl, "Recognition of handwritten script: A hidden Markov model based approach," in *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, pp. 928–931, 1988.

[Nag86] R. Nag, K. H. Wong, and F. Fallside, "Script recognition using hidden Markov models," in *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, pp. 2071–2074, 1986.

[Rab90] L. R. Rabiner, J. G. Wilpon, and F. K. Soong, "High performance connected digit recognition using hidden Markov models," in *Readings in Speech Recognition* (A. Waibel and K.-F. Lee, eds.), pp. 320–331, San Mateo, CA: Kaufmann, 1990.

[Sch91] L. R. B. Schomaker and E.-L. Teulings, "Stroke versus character-based recognition of on-line, connected cursive script," in *Third Int. Workshop on Frontiers in Handwriting Recognition*, (Chateau de Bonas, France), pp. 265–277, 1991.

[Tap82] C. C. Tappert, "Cursive script recognition by elastic matching," *IBM J. Res. Devel.*, vol. 26, pp. 765–771, Nov. 1982.