

확률 라이브러리 모델(PLM)에 의한 애너그램 문제 해결

강윤정¹, 이은석¹, 태강수², 장병탁³
서울대학교 인지과학 협동과정¹, 컴퓨터공학부²

Molecular Simulation of Anagram Problem Solving with PLM

Youn Jung Kang¹, Eun Seok Lee¹, Kang Soo Tae², Byoung-Tak Zhang³
Interdisciplinary Program in Cognitive Science, Seoul National University¹
School of Information Technology & Engineering, Jeonju University²
School of Computer Science & Engineering, Seoul National University³

요 약

본 연구는 애너그램(anagram) 문제해결 과제에서 발생하는 제약들(영어 철자연쇄 집합의 출현 확률과 단어 내 위치 확률)을 동시에 병렬적으로 만족시키는 확률 모델 학습과정을 보인다. 애너그램에 관한 많은 선행연구들은 이 문제해결 과정이 단순히 기호처리적인 층위뿐만 아니라 하위기호적(subsymbolic) 층위에서의 상향식 처리로 인해 일어남을 밝혀왔고, 주로 영어 철자의 연쇄체의 확률값을 이용해왔다. 본 연구는 확률 라이브러리 모델(Probabilistic Library Model)을 통해 애너그램 문제해결이 한 번씩 끝날 때마다 철자 연쇄체의 출현 및 위치 분포 확률이 어떻게 유연한 변화를 갖는지에 집중한다. 하나의 문제를 풀고 나면 본 모델은 그 전 문제를 풀었을 때의 상태 패턴으로부터 변화를 보인다. 이러한 분포 변화를 통해 하위기호적 활동의 영향이 문제해결에 있어서 학습구조의 유연한 변화에 중요한 영향을 끼친다는 점을 확인했다.

인지과학 영역에서, 인간의 마음에 대한 계산적 관점은 크게 기호주의와 연결주의로 나누어 볼 수 있다. 기호주의는 마음을 이산적인 기호적 요소들의 논리적 관계로 표상하는 기호적 정보처리 엔진으로 보며, 연결주의는 이와 달리 마음을 제약을 만족하는 탐색과 같은 기본적 연산이 동반된 통계적 추론 엔진으로 본다. 특히 연결주의는 그 출현 이후 인지처리과정들 중 대표적인 '학습'에 대해 계산론적 관점의 연구를 크게 확장시켜왔다. 이는 기호주의가 상위 인지과정의 기호처리에 대한 설명만을 만족시킬 수 있었던 반면, 연결주의는 암묵적이거나 의식 층위에서 내성으로 관찰되지 않는 하위기호적(sub-symbolic) 처리에 대해 납득할 만한 결과를 제시하였기 때문이다.⁷ 하위기호 층위 처리에 대한 계산론적 연구는 보통 그 상위 층위에 존재하는 기호를 구성하고 연결하는 하위층위 요소와 기제가 무엇인가에 집중되어왔다. 특히 최근 들어 운동기술 획득과 같은 암묵적이고 자동적인 행동에서부터 고도의 문제해결 처리에 이르기까지 폭넓은 영역에서, 인간의 학습구조가 의식 층위에서 접근하기 힘든 하위 층위의 기호망의 확률 분포 양상의 유연한 변화로 설명하는 연구가 크게 늘어나고 있다. 가장 큰 이유는 실제계에서 인간의 마음으로 들어오는 실제 데이터(자극)의 불확실성 때문이다. 극히 부분적인 정보에만 의존하여 몇 차례의 시행만으로 개념범주화에 성공

하거나, 정확히 일치하지 않는 과거의 문제 해결 경험에 유추를 적용해 새로운 문제를 해결하는 등의 인간 학습능력을 탐구하는 데 유연한 학습구조 변화를 주목하지 않을 수 없다. 본 연구는 이러한 인간의 학습구조 변화에 대한 확률 라이브러리 모델을 제시하고,⁸ 적용 문제로 애너그램을 택했다. 애너그램은 임의적으로 배열된 철자들의 순서를 재배치하여 실제 단어를 만들어내는 게임이다. 예를 들어 HCETA의 해결답은 TEACH 혹은 CHEAT가 된다. 애너그램 문제해결의 흥미로운 점은 이것이 고도의 의식적인 기호처리를 요구하는 과제임에도 불구하고, 하위기호적 층위의 처리가 필히 동반되어야 하기 때문이다.⁹

1. 애너그램 문제에 내재하는 제약조건들

일차적으로 애너그램 문제해결의 계산 모델의 기능은 철자 나열의 순서를 바르게 표상하는 것이 되어야 한다. 또한 하위기호적 층위에서의 문제해결을 다루는 모델이라면 영단어의 음성학적 통계치를 주로 이용하며 직접적인(intact) 어휘집 접속(lexicon access)은 지양해야 한다. 물론 애너그램 문제해결을 하는 데 장기기억에 저장된 단어의 직접적 인출을 부정할 수는 없지만 본 연구의 목적상 여기에서 다루지는 않을 것이다. 그렇다면 실제 존재하는 영단어에 내재하는 철자 배열 순서에

대한 학습 구조를 가장 크게 결정하는 제약들은 무엇일까.

아마도 에너지그림 문제해결에 관한 수많은 심리학 연구들 중에서 볼 수 있는 가장 직접적이고도 쉬운 제약조건은 두 개의 철자 연쇄체(이하 *bigram*)에 대한 높은 확률값일 것이다. (보통 철자구조적orthographic 접근이 많으나 본 연구에서는 음성학적 접근을 취한다.) *bigram*에 대한 제약 외에 세 개의 철자 연쇄체인 *trigram* 제약이 있을 수 있다. 본 연구에서 다루는 제약들을 나열하면 다음과 같다;

1. 특정 *bigram*의 출현 확률 분포 (*bigram affinity effect*)

2. 1의 *bigram*이 특정 단어 내에서 배열될 때 그 위치의 확률 분포 (*position effect*)

이 밖에도 여러 제약들을 들 수 있겠지만(예를 들어 주어진 철자들은 반드시 한 번에 한 곳에만 위치 가능한 제약을 들 수 있다) 1.과 2.의 확률 분포는 에너지그림 문제해결의 횟수가 증가하면서 변화 양상을 띠며, 그 양상이 문제해결에 대한 학습구조를 반영한다. 예를 들어 에너지그림 HCETA가 주어졌을 때, 문제해결자는 HC, CE, ET, TA, EA, CH 등등의 *bigram*들의 출현분포값을 자신의 학습구조에 내재하며, 문제해결이 거듭되면서 그 분포값을 암묵적으로 (비의식적으로) 조절하는 과정을 거쳐게 될 것이다. 예를 들어 숙련된 에너지그림 해결자라면 앞서의 문제에 EA와 CH *bigram* 출현확률이 높다는 것을 암묵적으로 지각할 것이며, 또한 EA는 한 단어 내 배열순서(1 2 3 4)에서 1번 혹은 2번에 올 확률이 높다는 것, 또 CH는 1번 혹은 4번에 올 확률이 높다는 것을 알 것이다. 초보적 해결자는 이런 구조에 대한 지식이 전혀 생성되지 않았을 것이다. 그러나 문제를 계속 해결해갈수록 이러한 학습구조는 변화를 겪을 것임은 직관적으로 분명하다.(그림 1) 학습구조는 의식적으로는 문제해결에 개입되지는 않지만 암묵적으로

	T	H	E	O	R	Y	
T	X	--HT	TE	TO	TR--	--TY	TR
H	--HT	X	HE	X	X	X	HT
E	X	X	X	EO--	--ER	EV	ER
O	X	X	X	X	OR--	OR	OR
R	X	X	RE--	X	X	--RY	RY
Y	X	X	X	X	X	X	X

그림 1 자극 THEORY에 대한 *bigram affinity*의 활성화 양상의 예 X는 전혀 활성화되지 않음을 나타내고, 나머지 *bigram*들은 크거나 진할수록 그 활성화 정도가 강함을 나타낸다.

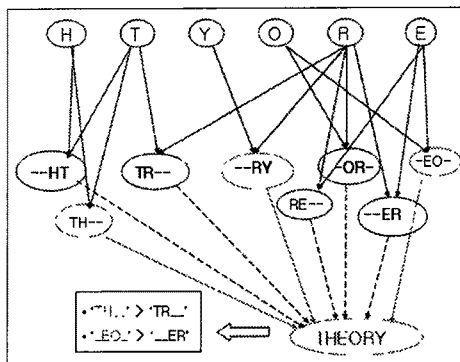


그림 2. 에너지그림 HTYORE 해결과정에 대한 개괄적 학습구조. TH와 TR에 대한 *bigram affinity*가 활성화된 후 정답 단어 THEORY가 제시된 후 *affinity*의 강도(하위기는 TH가 더 강화된다).

호층위에서) 제약조건 1과 2을 만족하여 문제해결에 개입한다.

또한 이 제약조건을 만족하는 것은 병렬적으로 이루어진다. Novick은 숙련된 에너지그림 해결자의 경우 순차적(sequential) 탐색보다는 병렬적 탐색에 의존하는 경향이 훨씬 뚜렷하게 나타남을 보이고 있다.² 연결주의 이론에서 “의식하지 못하는 사이에 문제해결이 가능해지는 현상에 대한 계산적 모델링이 가능” 하며, 이와 같은 모델링은 “하위기호적 층위에서의 초병렬적 계산처리를 통해 가능하다”는 논의도 활발하다.^{1,2,3,4,5,9}

Byoung-Tak Zhang의 확률 라이브러리 모델은 이러한 초병렬주의(Massive Parallelism)를 분자 층위의 알고리즘을 사용하여 구현하는 분자컴퓨팅 계산모델로서, 연결주의의 계산모델이 내세우는 하위기호 층위의 처리과정을 거의 실제적으로 처리할 수 있는 잠재성을 가지고 있다.⁸

2. 확률 라이브러리 모델

확률 라이브러리 모델은 분자 컴퓨팅을 사용하여 데이터로부터의 학습이 가능하다. 이 모델은 불확실한 실제 데이터로부터 학습을 가능하게 하고 예러에 비교적 민감하지 않은 결과를 추출할 수 있는 분자 컴퓨터 구축 가능성을 보이고 있다. DNA Computing은 대규모 병렬 프로세스를 구현하기에 가장 적합한 방법론으로 생각된다. 본 연구는 이러한 이유로 DNA Computing을 사용하였지만, 아주 적은 수의 기본적인 연산만을 적용했다. DNA Computing의 기술적 특징들을 잘 적용한다면 인지 과정의 시뮬레이션에 있어서 또다른 가능성을 제시할 것이라고 생각한다.

3. 학습 효과의 모델링

3.1. 알파벳 표상 시퀀스

26개의 각 알파벳을 표상하는 시퀀스를 인코딩하는 단계이다. 각 알파벳의 시퀀스는 다음과 같다.

- | | |
|----------------|---------------|
| a: GAAATGAGTT | n: TGATTATGTC |
| b: TGATGCTACA | o: ACATATCGGT |
| c: GGTGTGGCGG | p: CTCTTCAACG |
| d: CAGTTATTTTC | q: AGAAGAAACT |
| e: GGCTACGATG | r: CTGTCGAGTG |
| f: GAGCACTGCG | s: ACATTTACC |
| g: AGACGACGAC | t: ATCACGTAAT |
| h: GTAGAGGTAC | u: CAGAACGGAT |
| l: TCTCAGAGAT | v: TGTGTAGTGC |
| j: TCGATTTGGA | w: CGCGGCGATG |
| k: TAGGCAGATG | x: CTATGCTGGG |
| l: ACAGCACTAC | y: AACTTGGCGC |
| m: TCGGATAGGA | z: GCATGATCAT |

3.2. bigram affinity effect

규칙 시퀀스는 에너그램을 해결하는 인간의 머릿속에 들어 있을 것으로 가정되는 지식 베이스이다. 예를 들어 영어 단어에서 E와 A는 서로 연결 강도가 강하지만, W와 S는 연결 강도가 약하다. 규칙 시퀀스들은 이렇게 어떤 알파벳들이 어떤 알파벳과 서로 어떤 순서대로 연결이 잘 되어 있는지에 대한 지식을 나타낸다. 따라서, 이 규칙 시퀀스들이 적으면 적을수록 단어 형성은 어려움을 겪을 것이다. 에너그램에 익숙한 사람

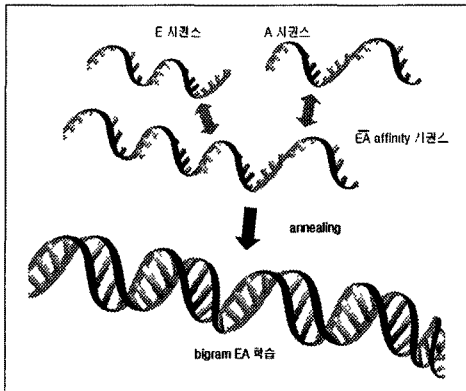


그림 3. 철자 E와 A가 자극으로 주어졌을 때 내재된 bigram EA가 활성화된다. 실제 DNA 컴퓨팅에서는 위의 세 시퀀스가 annealing을 통해 double strand가 형성되었을 때 활성화된 것으로 본다.

일수록 이 시퀀스들의 양이 많아질 것으로 추측된다.

규칙 시퀀스의 생성은 DNA 염기의 상보적 결합을 이용한 것으로, 기존의 알파벳 시퀀스의 상보적인 염기 서열을 사용한다. 즉 EA의 규칙을 나타내는 시퀀스는 E, A 시퀀스와 결합되어 그림 3과 같은 형태를 취한다.

이렇게 DNA 상보결합을 통해 형성된 double strand들은 온도를 높여 다시 single strand로 만든 후 PCR(Polimerase Chain Reaction)을 통해 지수증가적으로 그 수가 늘어난다. 비유적으로 이야기하자면 bigram EA에 대한 암묵적인 기억이 증가하는 것이다. 이러한 절차는 새로운 문제 집합이 들어올 때마다 반복된다. 그에 따라 bigram affinity의 확률값은 'AA'부터 'ZZ'까지 조금씩 변화하게 된다. 즉 머릿속에 내재되어 있던 bigram 출현 확률의 분포값이 계속해서 유연하게 변화하는 것이다.

3.3. 학습구조의 변화

에너그램 문제가 연이어 제시되었을 때 시간에 따른 학습구조의 변화를 보이고자 한다. 학습구조의 입력과 출력은 답으로서 제시될 가능성이 있는 모든 double strand들의 확률분포이다.

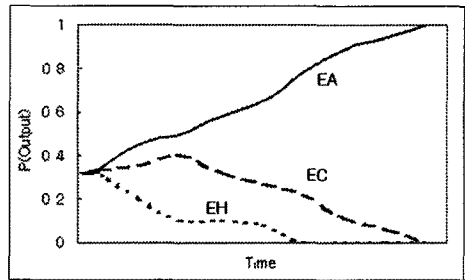


그림 4. 철자 E, A, C, H가 제시되었을 때 형성되는 bigram들과 이들이 문제 자극 제시의 반복에 따라 강화되는 양상

형식적으로, 특정 시간 t 에서 특정 학습구조의 입력과 출력의 상태 X_t 와 Y_t 는 각각 임의적 변수이다. 각 변수는 X_t 와 Y_t 에 대한 set size N_x 와 N_y 의 multinomial distribution으로부터의 값들 중 하나를 취한다. 예를 들어 X_t 는 에너그램 문제 H-E-T-A-C가 제시되었을 때 여기에 반응하여 활성화되는 bigram(EA나 CH 같은)들의 각각의 튜브 속 개수를 나타낼 수 있을 것이다. 즉 주어진 5개 철자들을 이어서 만들 수 있는 bigram들이 특히 많이 활성화될 것이다. 또 Y_t 는 철자들의 배열 순서를 나타낸다. 각각의 철자 스트림 순서의 set마다 개별적인 값을 갖는다. 앞의 예에서는 TEACH나 CHEAT가 형성되는 double strand들의 개수 등이 될 것이다. 또한 우리는 이 문제가 제시되고 그에 대한 출력이 나오는 매 단계 각각의 학습구조의 temporal dynamics를 모델링하고자 한다. X_t 와 Y_{t-1} 가 Y_t 를 결정하는 방식이 그 예가 될 수 있다. 특정 iteration d 동안 몇 개의 문제 set를 처리하는 것

을 $x = X_1 = X_2 = X_3 = \dots = X_t$ (특정 값 x 에 임의의 변수들을 대입하는 등의) 로 정의한다. 이 입력이 계속 진행된 후 특정 trial 단계에서 문제가 하나 주어졌을 때 우리는 이 학습 구조의 시간에 따른 진화 양상을 관찰할 수 있게 될 것이다.

입력과 출력 변수들 간 관계는 그림 6의 graphical model, dynamic belief network로 표현될 수 있다. 그림 6은 전형적이고 단순한 hidden Markov model이다. 이 모델의 전형적인

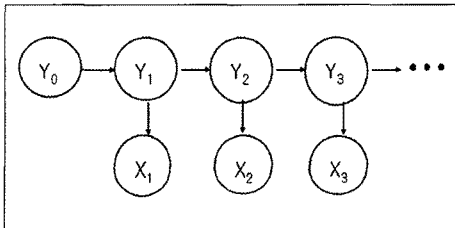


그림 184. 학습구조의 HMM 구현

사용양식은 연속적으로 개별적 입력들이 제시되지만 여기서는 입력들이 순서를 가지고 일정 단계마다 재투입된다. (예를 들어 t_0 에서 TEACH에 대한 문제가 입력되었다면 t_{100} , t_{200} 에서 이 문제는 다시 투입된다.) 또 보통 이 모델은 개별적인 hidden state들의 연쇄를 통해 transition을 이루지만 여기서는 적절한 증가추세를 가지고 단일 state로 수렴하는 양상을 보이고자 한다.

그림 6에서, X_t 와 Y_t 를 연결하는 화살표의 집합은 $P(X_t | Y_t)$ 에 상응하며, 이는 X_t 와 Y_t 사이의 effect transmission probability를 나타낸다. (예를 들면, bigram affinity에 해당하여 형성된 double strand의 분포 양상이 단어 형성에 끼치는 확률.) Y_t 와 Y_{t-1} 를 연결하는 화살표의 집합은 $P(Y_t | Y_{t-1})$ 에 상응하며, 이는 출력된 학습구조에 영향을 끼치는 기억을 나타낸다. $P(X | Y)$ 와 $P(Y | Y_{prev})$ 이 두 분포는 학습 구조를 결정하는 상태를 내포한다고 볼 수 있다.

bigram affinity 확률과 철자 배열순서 확률, 이 두 분포의 변화는 출력이 다시 이전의 구조로 입력되면서 그 모델링이 가능해진다. 즉 실험에서의 학습은 매 튜브에서 형성된 double strand들이 적절한 계수를 가지고 증폭된 후 다시 입력으로 다음 튜브로 복사되어 들어가면서 이루어진다.

4. 시뮬레이션: bigram affinity effect

본 연구의 시뮬레이션에 사용된 프로그램은 김준식 외에서 쓰인 것으로, 특정 DNA 시퀀스 간 결합 양상을 결합에너지와 온도에 따라 보여준다. 이 시뮬레이션은 실제 wet 실험의 결과와 거의 일치하는 반응을 보인다.¹⁰

이번 실험의 가설은 평소애 자주 접하는 단어의 특정 철자 세트에서 철자들 간의 친숙성이 높아짐을 보이는 것이다. 실험을 위해서 초기 라이브러리에는 영문 알파벳 bigram들이 AA에서 ZZ까지 676가지(26×26)가 있고, 초기 라이브러리를 학습시키기 위해서, 평소애 자주 접하는 blink, black,

blank, blind, blade와 자주 접하지 않는 xenic, xenia를 혼란 예제로 사용하였다. 실험 결과는 그림 7과 같다.

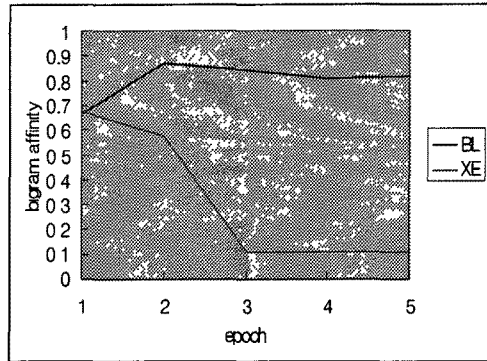


그림 185. bigram BL, XE의 특정 횟수 학습후 분포 변화

초기 라이브러리에 영문 철자 bigram은 천 개씩 모두 676000개가 존재하고, 라이브러리와 상보 결합하게 될 영문 단어들은 각각 천 개씩 7000개이다. 입력 단어가 우리가 자주 접하는 쉬운 단어라면, 예를 들어 blink일 경우 라이브러리에서 상보 결합을 하게 되면, 상보 결합에 참여한 bigram수를 세어서, 미리 정한 상수와 곱하고, 이 값을 최초 천 개와 더해 다음 라이브러리의 해당 bigram수로 정한다. 그리고 현재 라이브러리에서 상보 결합한 DNA수와 최대 가능한 상보 결합 DNA수와 비율을 계산한다. 곧 이 비율이 현재 라이브러리에서의 해당 bigram의 분포 값이면서 실제에서의 bigram 친숙성을 의미한다. 그림 7에서, 입력 단어가 일상생활에서 친숙한 단어일 때, 이 단어들과 자주 상보 결합하게 되는 bigram은 라이브러리 내에서 분포 값이 커짐을 볼 수 있다. 하지만, 입력 단어가 일상 생활에서 친숙하지 않은 단어, 예를 들어 xenic, xenia같은 경우는 이 단어와 상보 결합한 bigram 수를 세어 미리 정한 상수와 곱한 값을 라이브러리에서 제거한다. 그리고 이 값이 다음 라이브러리의 해당 bigram 분포 값에 반영된다. 위의 그림에서 보면, xe bigram은 분포 값이 줄어들음을 볼 수 있다. 자주 접하게 되는 단어에서 특정 bigram이 반복된다면, 이 bigram은 나머지 bigram보다 친숙성이 좋아지게 된다.

5. 결과 및 논의

확률 라이브러리 모델(PLM)을 통해 에너그램 문제해결에 대한 학습구조 변화를 기술하였다. 본 연구에서는 특정 bigram의 출현 확률 분포를 에너그램 해결시 필수적으로 동반되는 제약조건으로 삼았다. 이 제약조건의 확률 분포는 에너그램 문제해결의 횟수가 증가하면서 변화 양상을 띠며, 그 양상이 문제해결에 대한 학습구조를 반영한다. 시뮬레이션 결과는 이 제약조건을 만족을 통해 충분히 에너그램 학습 능력이 변화할 수 있음을 보이고 있다. 하나의 문제를 풀고 나면 본 모델은 그 전 문제를 풀었을 때의 상태 패턴으로부터 변화를 보였다. 문제는 이러한 패턴 변화를 문제해결자는 의식적으로 깨닫지

못한다는 데 있다. 이는 하위기호적 활동의 영향이 문제해결에 중요한 영향을 끼친다는 점을 암시한다. 대규모 병렬 처리가 하위 지각에 어떠한 영향을 끼치는지에 대해, 또 이 하위 지각 처리들이 상위 지각으로 연결되면서 어떤 제약조건들의 영향을 받는지에 대한 연구가 필요하다.

참고 문헌

1. David B. Grimes and Michael C. Mozer, "The Interplay of Symbolic and Subsymbolic Processes in Anagram Problem Solving," NIPS 2002.
2. Laura R. Novick and Steven J. Sherman, 'On the nature of insight solutions: Evidence from skill differences in anagram solution,' 'The Quarterly Journal of Experimental Psychology, 2003.
3. David Jacobs, et al., 'Fragment completion in humans and machines,' NIPS 2003.
4. Douglas R. Hofstadter, 'The architecture of Jumbo,' in *Fluid Concepts and Creative Analogies*, 1995.
5. Paul Smolensky, 'Information processing in dynamical systems,' in *Parallel distributed processing: Explorations in the microstructure of cognition*, 1986.
6. . 'Connectionist Modeling: Neural Computation / Mental Connections,' *Mind Design II*, 1997.
7. Byoung-Tak Zhang and Ha-Young Jang, 'A Bayesian Algorithm for In Vitro Molecular Evolution of Pattern Classifiers,' Preliminary Proceedings of the Tenth International Meeting on DNA Computing, pp. 294-303, 2004.
8. Byoung-Tak Zhang and Ha-Young Jang, 'Molecular Learning of wDNF Formulae,' 2005.
9. 이은석, 윤지은, 장병탁, 'DNAGram: Anagram 문제해결에 관한 분자 컴퓨팅 연구,' 한국인지과학회, 2003.
10. 김준식, 김중찬, 노영균, 이동윤, 장병탁, 'DNA 컴퓨팅 연산과정에 통계물리적 예측,' 정보과학회, 2005. (제출예정)