

## 컴퓨터는 ‘천재적인 바보’

경영학과 2008- 팽 동 은

### 1. 들어가며

많은 철학자, 과학자들은 인간의 지능이 고유한 것이어서 컴퓨터가 인간의 지능을 완벽히 흉내 낼 수 없다고 주장해왔다.

하지만 지구가 우주의 중심이 아니라는 사실이 알려졌을 때처럼, 컴퓨터가 인간의 지능을 빠른 속도로 따라잡고 있다는 사실을 인정하게 만드는 일들이 점점 빈번하게 일어나고 있다.

인간 뇌의 신비로 여겨져 온 바둑에서도 결국 컴퓨터 지능이 인간 지능을 따라잡는 일이 생겼다. 2013년 3월 20일, 프랑스의 바둑 소프트웨어 ‘crazy stone’이 넥 점을 깔고 일본의 이시다 요시오 9단에게 석 집을 이겼다. 이는 아마추어 5~6단 정도의 실력이다. 요시오 9단은 이 컴퓨터를 “냉정함과 유연함이 상당한 천재”라고 평가했다.

‘패’ ‘두터움’ ‘기세’ 등 이전까지 기계가 알 수 없는 영역이라고 생각되었던 부분에서도 컴퓨터의 침범이 생겨나기 시작한 것이다.

구글이 개발한 “신경망 알고리즘”도 파격적이다. crazy stone이 바둑에서 인간을 이기기 한 달 전 즈음인 2월 18일에는, 미국 IT 전문매체 wired는 구글이 모바일 운영체제 안드로이드 최신 버전에 “신경망 알고리즘”을 적용한 음성인식 시스템을 넣었다고 발표했다. 인간 두뇌가 반복 학습을 통해 새로운 지식을 얻고 습득하듯 이 신경망 알고리즘도 스스로 학습하는 구조로 설계돼 있다.

“신경망 알고리즘”을 활용해 컴퓨터는 1000만 장의 이미지 중 듣도 보도 못한 고양이 이미지를 골라내는 데 성공했다. 이 기계는 사전에 고양이가 무엇인지 학습한 적도 없고 고양이가 표시된 이미지를 사전에 본 적도 없음에도, 반복 학습을 통해 고양이 이미지를 분별해낸다. 생물학에서 뇌 속의 뉴런들이 특정한 물체를 인식하기 위해 하는 반복 훈련의 효과를 기술로 증명한 셈이다. 구글 창업자 래리 페이지가 2007년 “인간의 지능도 그렇게 복잡한 것이 아니어서 기계가 인간의 두뇌를 따라잡게 될 것”이라고 말한 지 6년 만이다.

최근 연달아 있었던 이 두 개의 사건은 컴퓨터가 인간 지능을 얼마나 많이 따라잡았는지 보여준다.

어떻게 기계에 불과한 컴퓨터가 인간의 지능을 흉내 내거나 앞설 수 있는 것일까? 컴퓨터가 인간 지능을 완전히 추월하면 인간이 로봇의 지배를 받게 되는 것 아닐까?

컴퓨터 구현의 원리와 알고리즘을 파악하면 컴퓨터 지능의 발달을 어떻게 생각하는 것이 옳은지, 어떻게 대처해야 하는지에 대한 힌트를 얻을 수 있다.

### 2. 컴퓨터는 ‘천재적인 바보’

현대의 컴퓨터는 무척 복잡해보이지만 사실은 간단한 원리들이 차곡차곡 계층을 더해 만들어진 ‘천재적인 바보’이다. 컴퓨터의 계층은 불 논리와 스위치에서 출발하여 최상위의 프로그래밍 언어로 이어지는데, 이 때 하드웨어와 소프트웨어 모두 하나의 기계로서 계층을 이룬다고 볼 수 있다.

컴퓨터의 계층 구조는 ‘기능적 추상화’의 원리와 밀접한 관련이 있다. 일단 논리 구성 블록으로 어떤 기능을 구현하게 만들어놓았으면, 그 다음부터는 그 블록의 세부 사항을 살필 필요 없이 반복적으로 사용할 수 있다. 이것이 컴퓨터 설계의 주요 본질 중 하나가 되는 ‘기능적 추상화’이다.

예를 들어, 어떤 입력신호가 들어왔을 때 불이 켜지는 장치를 만들어놓았다면 그 다음에는 어떻게

입력신호를 줄 것인지만 생각하면 된다. 버튼을 누름으로써 입력신호를 주고자 한다면 버튼이 입력신호로 연결되는 것만 생각하면 되고, 버튼이 불을 켜는 전 과정을 들여다볼 필요가 없다.

즉, 컴퓨터는 각 단계에서 주어진 일만 할 줄 아는 ‘바보’이지만, 정해진 일 하나는 기가 막히게 해내는 ‘천재’이다. 따라서 인간이 컴퓨터에게 쉬운 일부터 어려운 일까지 단계에 따라 적절하게 정해주면, 컴퓨터는 ‘천재적인 바보’의 역할을 해낼 수 있다.

그렇다면 이 ‘바보’를 ‘천재’로 만드는 것이 구체적으로 어떻게 이루어지는지, 즉 단순히 스위치를 까딱대는 것이 어떻게 복잡한 프로그램으로 이어지고 마침내 인간의 지능을 따라잡기까지 하는지 계층 별로 살펴보자.

### (1) 불 논리와 디지털 논리회로 - ‘바보’에게 생각하는 힘을!

불 논리와 디지털 논리회로는 컴퓨터 계층 구조의 가장 아랫단에 위치한다. 불 논리는 참과 거짓에 대한 대수이고, 디지털 논리회로는 컴퓨터가 불 논리에 따라 주어진 명제의 참 거짓을 판단할 수 있게 해준다.

불 논리를 컴퓨터로 표현할 수 있게 됐다는 것은 가만히 있으면 깡통과 다를 바 없는 컴퓨터에게 생각하는 힘이 생겼음을 의미한다. 불 논리를 사용하면 어떤 복잡한 논리 명제도 수학적으로 기술할 수 있고, 참과 거짓까지도 판별할 수 있기 때문이다.

컴퓨터에게 생각하는 힘을 불어넣은 것은 1940년 클로드 새넌이 “계전식 스위치 회로의 기호 해석”이라는 박사 논문을 발표하면서 가능해졌다. 이 논문에서 그는 스위치 회로를 직렬, 병렬로 연결함으로써 불 논리를 구현할 수 있다는 것을 증명하였다. 새넌의 회로에서 참과 거짓은 스위치의 열림과 닫힘으로 표현된다. 현대 컴퓨터에 사용되는 디지털 논리회로는 바로 이 새넌의 공식을 따르는 것이다.

스위치를 한 번씩 까딱까딱 하면서 어느 세월에도 복잡한 계산을 할 수 있겠는가 싶지만, 현대의 제품에 장착되는 스위치는 1초에 무려 몇 십억 번 이상 열고 닫을 수 있게 만들어져 있다.

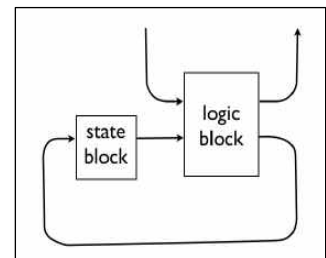
디지털 논리회로가 장착된 컴퓨터는 꽤 다재다능하다. 예를 들어 컴퓨터는 논리회로를 통해 사칙연산은 물론, 가위 바위 보에서 승자를 판단하는 것이나, 두 개가 다른지 같은지를 판단하는 것이 가능해진다.

뿐만 아니라, flip-flop이라고 불리는 특정한 유형의 회로를 만들어주면 컴퓨터에게 기억력이 생긴다. 현대 컴퓨터의 기억력은 인간과는 비교할 수 없을 정도로 그 능력이 뛰어나다. 예를 들어 100GB의 하드디스크에는 10만 권 이상의 책에 해당하는 정보를 넣을 수 있다. 인간은 그만큼의 지식을 완벽하게 기억하고 있기가 쉽지 않다.

### (2) 유한상태기계 - 천재가 되어가는 과정

논리회로와 메모리 기능을 사용하면 상태에 따라 다른 계산을 수행하는 ‘유한상태기계’를 만들 수 있다. 유한상태기계의 구성은 오른쪽과 같이 표현될 수 있다.

유한상태기계의 특징은 여러 개의 ‘상태’를 가질 수 있다는 것인데, 한 가지 상태만 가지면서 한 가지 계산만을 수행하는 기계는 나름 똑똑하기는 하나 만족할 수준은 못 된다. 컴퓨터에게 진정 똑똑하다는 수식어가 붙기 위해서는 조건에 따라 상태를 달리 하며 다양한 계산을 할 수 있어야 한다.



신호등은 유한상태기계의 좋은 예인데, 현재 신호등의 색깔(상태)을 기억하고 있다가, 건너기 단추가 눌리면서 새로운 입력 신호가 들어오면 출력 신호가 바뀐다. 즉 다른 색의 불이 들어온다. 다른 불이 들어온 이 상태를 다시 기억한다.

### (3) 소프트웨어 - 컴퓨터에게 언어를 가르치다

불 논리와 유한상태기계가 컴퓨터 하드웨어의 근간이 된다면, 프로그래밍 언어는 소프트웨어의 근간이 된다. 하드웨어와 소프트웨어는 크게 다를 것 같지만, 유무형의 차이만 있다고 봐도 무방할 정도로 소프트웨어는 기계적인 성격을 띤다.

하드웨어만 갖춘 유한상태기계는 0이라는 말을 들으면 스위치를 열고, 1이라는 말을 들으면 스위치를 닫는 것밖에 못한다. 그런데 수많은 스위치를 일일이 찾아다니면서 이걸 열고 저걸 닫아, 라고 명령하는 것은 여간 귀찮은 일이 아니다.

반면 컴퓨터에게 프로그래밍 언어로 짜인 소프트웨어를 주면, 여러 단순한 행동을 한 단어로 정의함으로써 복잡한 명령을 한 번에 하달할 수 있다. 컴퓨터가 더욱 똑똑해지는 것이다. 소프트웨어가 어떻게 그런 힘을 갖는지 좀 더 자세히 살펴보자.

프로그래밍 언어는 '1번 스위치 열고 2번 스위치 닫아'를 '블루'라는 한 단어로 정의할 수 있다. 그리고 다시 '블루 실행하고 3번 스위치 닫아'를 '레드'라고 정의할 수도 있다. 이 경우 컴퓨터에 '레드'라는 명령이 떨어지면 컴퓨터는 '레드'를 해석하고, 다시 '블루'를 해석해서 최종적으로 1번 스위치 열고 2번 스위치 닫고 3번 스위치 닫는 행동을 한다. 스위치 세 개를 조작하는 이 복잡한 행동을 '레드'라는 한 단어로 정의할 수 있는 것이다. 이것이 '서브루틴(subroutine)'의 원리이다.

서브루틴은 컴퓨터를 '천재'로 만드는 아주 강력한 특성이다. 서브루틴을 통해 클릭 한 번에 컴퓨터가 수천, 수만 가지 행동을 하도록 만들 수도 있다. 과연 천재라 할만하다.

또 다른 능력으로는 '재귀(recursion)'라는 것이 있다. 만약 '그린'이라는 명령이 '블루 한 번 실행하고 그린'이라는 의미를 가진다고 해보자. 그러면 '그린'을 실행하는 컴퓨터는 '블루를 한 번 실행하고, 또 블루를 한 번 실행하고, 또 블루를...'이런 식으로 '그린'을 무한 반복할 것이다.

재귀 기능을 통해 프로그래머는 컴퓨터에게 한 번의 명령으로 같은 행동을 무한하게 혹은 유한한 횟수만큼 반복시킬 수 있다. 위의 예에서 컴퓨터는 '그린'이라는 명령을 무한하게 반복하게 되는데, 이것이 대부분의 컴퓨터 프로그램의 속성 가운데 하나인 무한 루프이다. 무한 루프를 활용하면 컴퓨터에게 끊임없이 명령을 내릴 필요 없이 한 번만 명령하면 된다. 한 번만 말하면 그 후로는 알아서 척척 쉽 없이 계산을 하니, 몇 번을 말해도 못 알아듣는 인간보다 어떤 면에서는 낫다고 할 수도 있겠다.

그럼 프로그래밍 언어를 0과 1로 번역하는 일은 누가 대신 해주는 것일까? 놀랍게도 번역 또한 컴퓨터가 알아서 한다. 컴퓨터는 자신이 필요한 번역을 스스로 하는데, 특정 명령을 지시 받은 컴퓨터는 디렉토리에서 그 명령어 이름을 찾은 후에 해당하는 기계어 열을 찾아 수행한다.

따라서 프로그래머는 자신이 활용하는 언어가 어떤 절차를 거쳐 유한상태기계를 움직이시는는 걱정하지 않아도 된다. 앞에서 언급한 '기능적 추상화'라는 단어가 위의 원리를 잘 설명해준다. 이 원리에 의해 프로그래머는 프로그래밍 단계에만 충실하면 될 뿐, 자신이 일일이 마이크로프로세서를 현미경으로 들여다보며 회로가 잘 연결되었는지 확인할 필요가 없다.

### 3. 모방의 천재 - 컴퓨터의 보편성

아름다운 클래식을 만드는 사람도 천재지만, 그것을 그대로 연주하는 사람도 천재이다. 즉 창조에

강한 천재가 있는가 하면 모방에 강한 천재도 있다. 인간과 컴퓨터를 비교하자면 상대적으로 인간은 전자에, 컴퓨터는 후자에 가깝다.

우리가 일상적으로 다루는 컴퓨터를 보편 컴퓨터라고 하는데, 이 컴퓨터는 다른 컴퓨터를 완벽히 모방하는 재주가 있다. 한마디로 임의의 두 컴퓨터 사이의 성능 차이는 속도, 메모리 크기 외에는 없다고 보아도 무방하다. 이론적으로 아주 뛰어난 컴퓨터는 더 빠르고 효율적으로 계산을 수행할 수 있지만, 그렇다고 해서 50년 전의 골동품 컴퓨터가 계산하지 못했던 문제를 풀 수 있는 것은 아니다.

컴퓨터가 모방의 천재에 그치는 것은 안타까운 일이다. 왜냐하면 하나의 컴퓨터가 풀지 못하는 문제는 다른 어떠한 컴퓨터도 풀지 못하기 때문이다. 이러한 문제들 중에는 멈춤 문제, 수리 명제 자동 판별 문제와 같은 유명한 문제들이 있다. 이 문제들은 컴퓨터가 풀기 어려운 것이 아니라, 불가능하다고 결론지어진 문제들이다.

그러나 이것이 컴퓨터보다 인간이 뛰어나다는 것을 증명하는 근거가 되지 못한다. 지금까지 알려진 한, 인간이 증명할 수 있는 정리는 컴퓨터도 증명할 수 있다. 또한 컴퓨터가 계산하지 못하는 문제는 인간도 계산할 수 없다.

오히려, 오직 한 종류의 보편 컴퓨터만이 존재한다는 가정은 인간의 뇌를 컴퓨터가 모방할 수도 있다는 가능성을 보여준다. 만약 인간의 뇌가 하는 계산도 물질적인 방법으로 이루어진다면, 그 계산 과정을 적절히 프로그래밍 함으로써 보편컴퓨터도 인간 뇌의 사고방식을 흉내 낼 수 있기 때문이다.

한편, 이러한 한계를 뛰어넘는 가장 유력한 대안으로 양자 컴퓨터가 떠오르고 있다. 양자 컴퓨터에서는 메모리에 1이나 0 둘 중 하나만 저장하는 것이 아니라 1과 0을 동시에 저장하는 것이 가능하다. 양자 역학을 응용해 한 단계 위의 컴퓨터가 등장한다면 컴퓨터 역사가 새로 쓰일 정도로 센세이셔널 할 것이나, 현재로서는 그 성능이 디지털 컴퓨터의 한계를 뛰어넘는지, 아니면 속도만 빠른 또 하나의 보편 컴퓨터일 뿐인지에 대해 불확실한 상태이다.

#### 4. 알고리즘과 휴리스틱 - ‘고지식한 천재, 눈치 빠른 천재’

알고리즘은 위에서 다룬 컴퓨터 구현의 원리와는 다소 다른 주제로서, 컴퓨터 지능의 범위와 가능성을 보여주는 또 하나의 중요한 요소이다.

알고리즘은 특정 목표를 “완벽하게” 달성할 수 있는 보장된 절차이다. 컴퓨터 알고리즘은 일반적으로 프로그램으로 표현되는데, 하나의 알고리즘을 어떤 프로그래밍 언어로도 표현할 수 있다. 또한 소프트웨어에서만 아니라 하드웨어에서도 레지스터와 논리블록을 이용하여 알고리즘을 구현할 수 있다. 즉, 알고리즘은 컴퓨터가 하드웨어부터 소프트웨어까지 모든 단계에서 사용하는 ‘문제 풀이 방식’이다.

하나의 목표 달성을 위해 다양한 알고리즘이 사용될 수 있기 때문에, 다양한 알고리즘 중 가장 빠른 것을 찾는 것은 프로그래머의 중요한 과제이다. 알고리즘의 속도는 단순히 답을 도출하는 데 걸린 시간으로 측정하지 않는다. 같은 알고리즘이라도 문제의 크기에 따라 걸리는 시간이 달라지기 때문이다. 대신 알고리즘의 속도는 문제의 크기가 커짐에 따라 소요 시간이 얼마나 늘어나느냐로 정해진다. 한마디로, 입력이 커지면 “결국 어떻게 될 것인지”를 따지는 것이다.

다양한 알고리즘 대안 중 알려진 최상의 알고리즘 속도가  $2^n$ 과 같이 지수로 표현되는 문제들이 있다. 이 경우 문제(n)가 조금만 커져도 소요 시간이 엄청나게 늘어나기 때문에, 문제가 크면 현실적인 시간 안에 완벽하게 답을 내는 것이 불가능하다.

이러한 문제들의 경우, 컴퓨터는 고지식하게 완벽을 추구하지 않고 휴리스틱을 사용함으로써 소요

시간을 크게 줄인다. 휴리스틱은 만점이 아닌 합격점에서 만족하는 방식이다. 알고리즘은 '실패 없이' 달성하는 절차임에 반해, 휴리스틱은 '거의 실패하지 않고' 달성하는 절차라고 할 수 있다.

체스 챔피언을 이긴 천재 컴퓨터들도 모두 휴리스틱을 이용해서 이긴 것이다. 체스 게임의 시작부터 끝까지 모든 경우의 수를 고려한다면 그 게임에서 100% 이길 수 있지만, 그러기에는 시간이 너무 오래 걸린다. 휴리스틱을 사용하기 때문에 컴퓨터는 종종 악수(惡手)를 유발하기도 하고, 인간에게 패배하기도 한다. 계산해야 하는 경우의 수가 많아질수록 인간이 유리하다. 체커 - 체스 - 바둑 순으로 컴퓨터가 인간의 지능을 따라잡고 있는 것도 이러한 이유에서라고 볼 수 있다.

### 5. 나가며 - 인간 지능 vs. 컴퓨터 지능

컴퓨터는 '천재적인 바보'라는 별명이 잘 어울린다. 정해진 일만 곧이곧대로 하는 한편, 정해진 일에 대해서는 완벽하게 일처리를 해내기 때문이다. 바보같이 아주 단순한 일을 차근차근 모아 복잡한 일을 천재적으로 해내는 것을 통해서도 컴퓨터의 특징이 드러난다.

이 '천재적인 바보'의 학습 속도는 엄청나다. 최근에는 인간이 차별적인 강점을 갖던 휴리스틱, 즉 넘겨짚기 분야에도 컴퓨터가 본격적으로 발을 들여놓기 시작했다. 컴퓨터가 휴리스틱을 이용하여 체스나 바둑, 퀴즈, 요리 등의 분야에서 인간을 압도하는 것을 보면 컴퓨터의 천재성이 무서운 속도로 인간의 지능과 사고방식을 따라잡고 있다는 것을 알 수 있다.

아름다운 음악, 아름다운 그림도 인간만이 만들 수 있다고 여겨져 왔지만, 최근 인기를 끌고 있는 모바일 게임 '음악 마법사'는 몇 가지 음을 쳐 놓으면 컴퓨터가 아름다운 멜로디와 박자, 화음을 곁들여 멋진 음악을 탄생시켜준다.



왼쪽의 그림은 컴퓨터가 그린 그림이다. 이 그림은 컴퓨터가 위상최적설계 알고리즘에 따라 스스로 그림을 그린 것이다. 이러한 미술을 변분미술이라고 한다.

컴퓨터 지능의 발달은 애써 무시할 수 있는 수준을 넘어섰다. 철학자들의 우려대로 컴퓨터가 인간의 뇌를 정확히 흉내 내는 날이 올 수도 있다.

이런 상황에서, 컴퓨터와 아등바등 영역 싸움을 벌이는 것은 무의미하다. 컴퓨터에게 내어줄 부분은 과감히 내어주면서 인간 고유의 영역을 더욱 확장시켜 나가야 한다. 기계가 요리까지 맡아서 하게 됐다면, 이제 요리라는 인간의 고유 영역을 컴퓨터에게 내어주고 한 발 나아가면 된다. 컴퓨터가 요리를 하는 덕분에

에 인간은 그 시간 동안 더 생산성 있는 일을 할 수 있게 되는 것이다.

아직 컴퓨터가 따라오지 못한 인간의 영역들이 많이 있다. 감수성, 오감, 패턴 인식, 상황 파악, 중요도 판단, 호기심 등등.

컴퓨터가 할 수 있는 일들은 컴퓨터에게 맡겨 놓고, 맡길 수 없는 일들에 집중함으로써 인간이 늘 한 발씩 앞서나가는 것이야말로 인간 지능의 확장이다. 그렇게 확장된 지능이 다시 컴퓨터에 의해 따라잡혔을 때에는, 인간의 지능은 이미 한 발 앞서 있을 것이다.

컴퓨터 지능의 발전을 두려워하거나 꺼려할 이유는 없다. 애써 부인할 필요도 없다. 성경의 이사야서에는 이런 구절이 있다. "진흙이 토기장이에게 '너는 도대체 무엇을 만들고 있는 거냐?'하고 말할 수 있겠으며, 네가 만든 것이 너에게 '그에게는 손이 있으나마나다!' 하고 말할 수 있겠느냐?"

컴퓨터가 어느 날 인간에게 물어본다. '주인님은 손이 있으나마나 입니다.' 이런 날이 올까? 알 수 없다. 하지만, 그 날에도 인간이 컴퓨터의 주인임에는 변함이 없을 것이다.