SNU 046.016 컴퓨터과학이 여는 세계 (Computational Civilization) Part IV

Prof. Kwangkeun Yi

Department of Computer Science & Engineering

차례

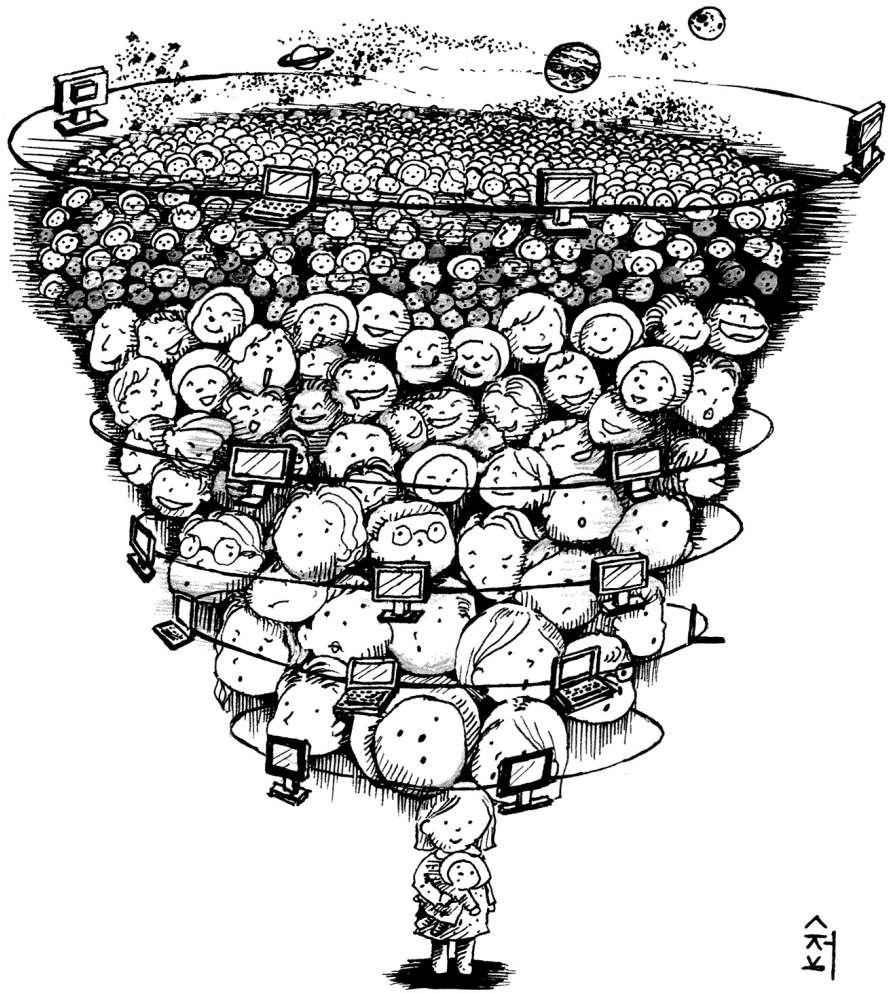
이전

이전

이전

다음

컴퓨터과학의 응용



인간 지능/본능/현실의 확장

컴퓨터과학의 응용 I: 인간 지능의 확장

기계는 압도할 것이다

컴퓨터과학의 응용 I: 인간 지능의 확장

기계는 압도할 것이다

- 인간의 근육을 압도했듯이 (산업혁명)

컴퓨터과학의 응용 I: 인간 지능의 확장

기계는 압도할 것이다

- 인간의 근육을 압도했듯이 (산업혁명)
- 인간의 지능을 압도할 것이다 (지능혁명)

컴퓨터과학의 응용 I: 인간 지능의 확장

기계는 압도할 것이다

- 인간의 근육을 압도했듯이 (산업혁명)
- 인간의 지능을 압도할 것이다 (지능혁명)
- 그리고 가속될 것이다
  - 컴퓨터의 한계/기록은 늘 깨질것이다
  - 현재의 컴퓨터는 "튜링소년"의 정의였을뿐

컴퓨터과학의 응용 I: 인간 지능의 확장

기계는 압도할 것이다

- 인간의 근육을 압도했듯이 (산업혁명)
- 인간의 지능을 압도할 것이다 (지능혁명)
- 그리고 가속될 것이다
  - 컴퓨터의 한계/기록은 늘 깨질것이다
  - 현재의 컴퓨터는 "튜링소년"의 정의였을뿐

문명 발전의 발자취

- 지능이 필요한 일을 기계에 맡겨간 과정
- 거부감은 늘 일시적이었을 뿐

컴퓨터과학의 응용 I: 인간 지능의 확장

기계는 압도할 것이다

- 인간의 근육을 압도했듯이 (산업혁명)
- 인간의 지능을 압도할 것이다 (지능혁명)
- 그리고 가속될 것이다
  - 컴퓨터의 한계/기록은 늘 깨질것이다
  - 현재의 컴퓨터는 "튜링소년"의 정의였을뿐

문명 발전의 발자취

- 지능이 필요한 일을 기계에 맡겨간 과정
- 거부감은 늘 일시적이었을 뿐

인간은 늘 기계와 팀이 되어

- 상상밖의 일을 이루며 문명을 새롭게 연출해왔다

기계가 압도하는 인간의 지능

기계가 압도하는 인간의 지능

- 세탁기, 상하수도, 증기기관, 내연기관

기계가 압도하는 인간의 지능

- 세탁기, 상하수도, 증기기관, 내연기관
- 전화교환기, 기차 자동운전, 로봇청소기

기계가 압도하는 인간의 지능

- 세탁기, 상하수도, 증기기관, 내연기관
- 전화교환기, 기차 자동운전, 로봇청소기
- 비행기 자동운전, 자동차 자동운전, 무인비행기, 무인자동차

기계가 압도하는 인간의 지능

- 세탁기, 상하수도, 증기기관, 내연기관
- 전화교환기, 기차 자동운전, 로봇청소기
- 비행기 자동운전, 자동차 자동운전, 무인비행기, 무인자동차
- 체스(1997, DeepBlue), Crosswords(1999, Proverb), Jeopardy!(2010, Watson), 바둑(2016, AlphaGo)

기계가 압도하는 인간의 지능

- 세탁기, 상하수도, 증기기관, 내연기관
- 전화교환기, 기차 자동운전, 로봇청소기
- 비행기 자동운전, 자동차 자동운전, 무인비행기, 무인자동차
- 체스(1997, DeepBlue), Crosswords(1999, Proverb), Jeopardy!(2010, Watson), 바둑(2016, AlphaGo)
- 영화추천, 책추천, 광고, 요리법고안, 미술품분류

기계가 압도하는 인간의 지능

- 세탁기, 상하수도, 증기기관, 내연기관
- 전화교환기, 기차 자동운전, 로봇청소기
- 비행기 자동운전, 자동차 자동운전, 무인비행기, 무인자동차
- 체스(1997, DeepBlue), Crosswords(1999, Proverb), Jeopardy!(2010, Watson), 바둑(2016, AlphaGo)
- 영화추천, 책추천, 광고, 요리법고안, 미술품분류
- 지식의 양, 지식의 탐색

기계가 압도하는 인간의 지능

- 세탁기, 상하수도, 증기기관, 내연기관
- 전화교환기, 기차 자동운전, 로봇청소기
- 비행기 자동운전, 자동차 자동운전, 무인비행기, 무인자동차
- 체스(1997, DeepBlue), Crosswords(1999, Proverb), Jeopardy!(2010, Watson), 바둑(2016, AlphaGo)
- 영화추천, 책추천, 광고, 요리법고안, 미술품분류
- 지식의 양, 지식의 탐색
- 병진단, 보험설계, 자산관리, 회계관리, 위법판정

기계가 압도하는 인간의 지능

- 세탁기, 상하수도, 증기기관, 내연기관
- 전화교환기, 기차 자동운전, 로봇청소기
- 비행기 자동운전, 자동차 자동운전, 무인비행기, 무인자동차
- 체스(1997, DeepBlue), Crosswords(1999, Proverb), Jeopardy!(2010, Watson), 바둑(2016, AlphaGo)
- 영화추천, 책추천, 광고, 요리법고안, 미술품분류
- 지식의 양, 지식의 탐색
- 병진단, 보험설계, 자산관리, 회계관리, 위법판정

우리 어떻해요?

기계가 압도하는 인간의 지능

- 세탁기, 상하수도, 증기기관, 내연기관
- 전화교환기, 기차 자동운전, 로봇청소기
- 비행기 자동운전, 자동차 자동운전, 무인비행기, 무인자동차
- 체스(1997, DeepBlue), Crosswords(1999, Proverb), Jeopardy!(2010, Watson), 바둑(2016, AlphaGo)
- 영화추천, 책추천, 광고, 요리법고안, 미술품분류
- 지식의 양, 지식의 탐색
- 병진단, 보험설계, 자산관리, 회계관리, 위법판정

우리 어떻해요?

기계와 팀이 되라!

인간 고유의 지능?

팀웍에 필요한, 순도를 높여야 할 인간만의 지능은 뭘까?

인간 고유의 지능?

팀웍에 필요한, 순도를 높여야 할 인간만의 지능은 뭘까?

- 더 나은 것을 상상하는 능력

인간 고유의 지능?

팀웍에 필요한, 순도를 높여야 할 인간만의 지능은 뭘까?

- 더 나은 것을 상상하는 능력
- 현재를 회의하고 의심하는 능력

인간 고유의 지능?

팀웍에 필요한, 순도를 높여야 할 인간만의 지능은 뭘까?

- 더 나은 것을 상상하는 능력
- 현재를 회의하고 의심하는 능력
- 질문하는 능력

인간 고유의 지능?

팀웍에 필요한, 순도를 높여야 할 인간만의 지능은 뭘까?

- 더 나은 것을 상상하는 능력
- 현재를 회의하고 의심하는 능력
- 질문하는 능력
- 질문이 왜 중요한지 설명하고 설득하는 능력

인간 고유의 지능?

팀웍에 필요한, 순도를 높여야 할 인간만의 지능은 뭘까?

- 더 나은 것을 상상하는 능력
- 현재를 회의하고 의심하는 능력
- 질문하는 능력
- 질문이 왜 중요한지 설명하고 설득하는 능력
- 관련없는 것들 사이의 관련성을 보는 능력

컴퓨터과학의 응용 I: 인간 지능의 확장

컴퓨터 없이는 불가능했던, 지식의 표현/생성/검색

- 지식 표현: 소프트웨어로 주고받는 전문지식
- 지식 생성: 기계학습
- 지식 검색: 구글 랭킹
- 팀웍지능과 군중지능

인간 지능의 확장 1: 지식 표현(1/4)

인류의 발전과 함께 해 온 지식 표현 방식

	방식	시절	도구
서술형(descriptive)	지식 표현	과학이전	언어
방정식형(equational)	지식 표현	과학이후	수학
계산형(computational)	지식 표현	컴퓨터이후	컴퓨터

인간 지능의 확장 1: 지식 표현(2/4)

서술형(descriptive) 지식 표현: "문과식", 말로 표현하기

"해는 그 몸이 지구보다 여러 갑절 크고, 그 바탕은 불이며, 그 빛깔은 붉다. 바탕이 불로 된 까닭에 그 본성은 온난하고, 빛깔이 붉기 때문에 그 빛은 밝다. 그 불꽃은 사방에 퍼져 환히 비치는데, 멀수록 점점 약해지지만, 그 거리는 수천만 리에 이른다."

- 홍대용, [의산문답], 1766년

"달이 가진, 매혹적인 둥근 기운은 지구까지 다다르고 지구의 물을 사로잡는다. 달이 최고점을 지나 빠르게 날아갈 때, 그 때 물이 같은 속도로 따라가지 못하면 바다물은 열렬한 기운에 휩싸인 곳에선 서쪽으로 쏠리게 된다."

- 케플러, [신 천문학], 1609년

인간 지능의 확장 1: 지식 표현(3/4)

방정식형(equational) 지식 표현: "이과식", 수학으로 표현하기

$$F = G\frac{m_1 m_2}{r^2}$$ 어디나 있는 끄는 힘

$$E = mc^2$$ 질량은 에너지

$$\Phi(x) = \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$ 종모양으로 분포하는 확률

$$[\![e]\!] = \bigsqcup_{i\geq 0} F^i(\bot)$$ 컴퓨터 프로그램의 의미

$$\alpha \circ F \sqsubseteq \hat{F} \circ \alpha$$ 안전한 프로그램 분석

$$\frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + rS\frac{\partial v}{\partial S} + \frac{\partial V}{\partial t} - rV = 0$$ 파생상품 가격의 변동

$$\nabla \cdot E = 0 \qquad \nabla \cdot H = 0$$

$$\nabla \times E = -\frac{1}{c}\frac{\partial H}{\partial t} \qquad \nabla \times H = \frac{1}{c}\frac{\partial E}{\partial t}$$ 하나로 춤추는 전기와 자기

인간 지능의 확장 1: 지식 표현(4/4)

계산형(computational) 지식 표현: 컴퓨터 소프트웨어로 표현하는 지식

인간 지능의 확장 1: 지식 표현(4/4)

계산형(computational) 지식 표현: 컴퓨터 소프트웨어로 표현하는 지식

- 물리학 지식: 방정식의 답을 근사계산하는 소프트웨어. 더 이상 물리방정식을 손으로 풀 수 없다.
  - 현대 물리학에서 약 5% 정도만 손으로 풀 수 있을 것

인간 지능의 확장 1: 지식 표현(4/4)

계산형(computational) 지식 표현: 컴퓨터 소프트웨어로 표현하는 지식

- 물리학 지식: 방정식의 답을 근사계산하는 소프트웨어. 더 이상 물리방정식을 손으로 풀 수 없다.
  - 현대 물리학에서 약 5% 정도만 손으로 풀 수 있을 것
- 수학 지식: 증명을 표현한 소프트웨어. 컴퓨터가 검산. 더 이상 수학증명을 손으로 검산할 수 없다.
  - Four-color theorem 증명: 2003년 컴퓨터가 검산완료 (Coq 시스템)
  - Feit-Thompson theorem 증명: 1963년 255페이지 증명; 2013년 컴퓨터가 검산완료(Coq 시스템)
  - Kepler conjecture 증명: 2005년 121페이지 증명(불확실); 2014년 8월 컴퓨터가 검산완료(Isabelle 시스템과 HOL 시스템)

SW로 표현한 Feit-Thompson theorem 증명 일부

```
Inductive in_image T R (D : T -> Type) (f : T -> R) (a : R) :=
  InImage (x : T) (x_in_D : D x) (a_is_fx : equal R a (f x)).

Inductive finite_of_order T (D : T -> Type) (n : natural) :=
  FiniteOfOrder (rank : T -> natural)
    (rank_injective : injective_in T natural D rank)
    (rank_onto :
       forall i, equivalent (less_than i n) (in_image T natural D rank i)).

(* Elementary group theory *)

Inductive group_axioms T (mul : T -> T -> T) (one : T) (inv : T -> T) :=
  GroupAxioms
    (associativity : forall x y z, equal T (mul x (mul y z)) (mul (mul x y) z))
    (left_identity : forall x,     equal T (mul one x) x)
    (left_inverse  : forall x,     equal T (mul (inv x) x) one).

Inductive group T mul one inv (G : T -> Type) :=
  Group
    (G_closed_under_mul : forall x y, G x -> G y -> G (mul x y))
    (one_in_G           : G one)
    (G_closed_under_inv : forall x, G x -> G (inv x)).

Inductive subgroup T mul one inv (H G : T -> Type) :=
  Subgroup
    (H_group    : group T mul one inv H)
    (H_subset_G : forall x, H x -> G x).

Inductive normal_subgroup T mul one inv (H G : T -> Type) :=
  NormalSubgroup
    (H_subgroup_G     : subgroup T mul one inv H G)
    (H_is_G_invariant : forall x y, H x -> G y -> H (mul (inv y) (mul x y))).

Inductive commute_mod T mul (x y : T) (H : T -> Type) :=
  CommuteMod (z : T)
    (z_in_H    : H z)
    (xy_eq_zyx : equal T (mul x y) (mul z (mul y x))).

Inductive abelian_factor T mul one inv (G H : T -> Type) :=
  AbelianFactor
    (G_group        : group T mul one inv G)
```

인간 지능의 확장 2: 지식 생성(1/2)

새로운 지식을 만드는 방식

- 디덕(deduction): "반드시 이끌기" (확실)

$$\frac{A \Rightarrow B \qquad A}{B}$$

- 앱덕(abduction): "원인 짐작하기" (아마도)

$$\frac{A \Rightarrow B \qquad B}{A}$$

- 인덕(induction): "짐작해서 이끌기" (아마도)

$$\frac{\text{관찰한 바} \quad A\text{일땐 늘 } B}{A \Rightarrow B}$$

기계 학습

인덕(induction)과 앱덕(abduction)

- 번역하기, 자동차운전하기
- 사진정리, 바둑두기, 축구, 당구
- 실력을 시험점수로, 성격을 읽는책으로
- 직급 적합성을 페북활동으로, 디자인을 스케치로, 좋아하나를 카카오로
- 미더운사이(reputation scoring)

기계 학습

인덕(induction)

- 특수 사실에서 보편 지식으로 건너뛰기
- 특수 정답예시(데이터)들로 부터 보편 답안(소프트웨어)을 짐작해내기
  - 새로운 프로그래밍 방법
  - 언어와 논리로 짤 수 없던 소프트웨어를 만드는 방법

특히, 깊은 신경망(deep neural net)의 인덕은:

- 답안(소프트웨어) = 깊은 신경망
- 예시(데이터) = 입출력 정답

기계학습 기술이 넘어야 할 허들

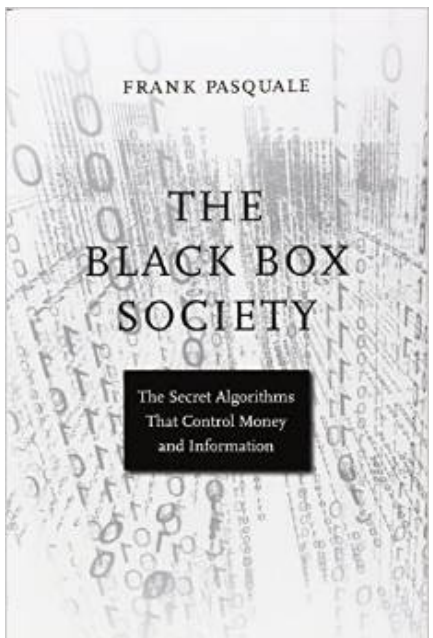
- 늘 부족한 데이터
  - 더 잘 배우려면 더 많은 데이터가 항상 기하급수로 더 필요
- 관찰한 데이터에 과도하게 맞추면
  - 오버해서 데이터에 맞추지 말기
  - 비유: 완전한 학점의 학생?
- 다다익선이어야
  - 학습량이 많으면 진실함수에 가까워지는
  - 그런 확률이 커지는 실용적인 방법이어야

기계학습의 함정

앱덕의 헛점

- 불완전한 인과관계 모델
  - 모델못한, 몰랐던 인과관계는 항상 존재한다
- 간과하는 미세한 데이터
  - 데이터의 홍수에 휩쓸리는, 작지만 중요한 시그날

읽어보기:



인간 지능의 확장 2: 지식 생성(2/2)

컴퓨터없이는 불가능했던 지식
생명체의 대규모 데이터를 판독/비교/분석.

- Human Genome Project: 인간 유전자 염기서열 규명 (2003년 4월 완성)
- Human Connectome Project: 인간 뇌 뉴런 구조 규명 (진행중)
- Big Mechanism Project: 거대한 시스템 자동 파악 (진행중)

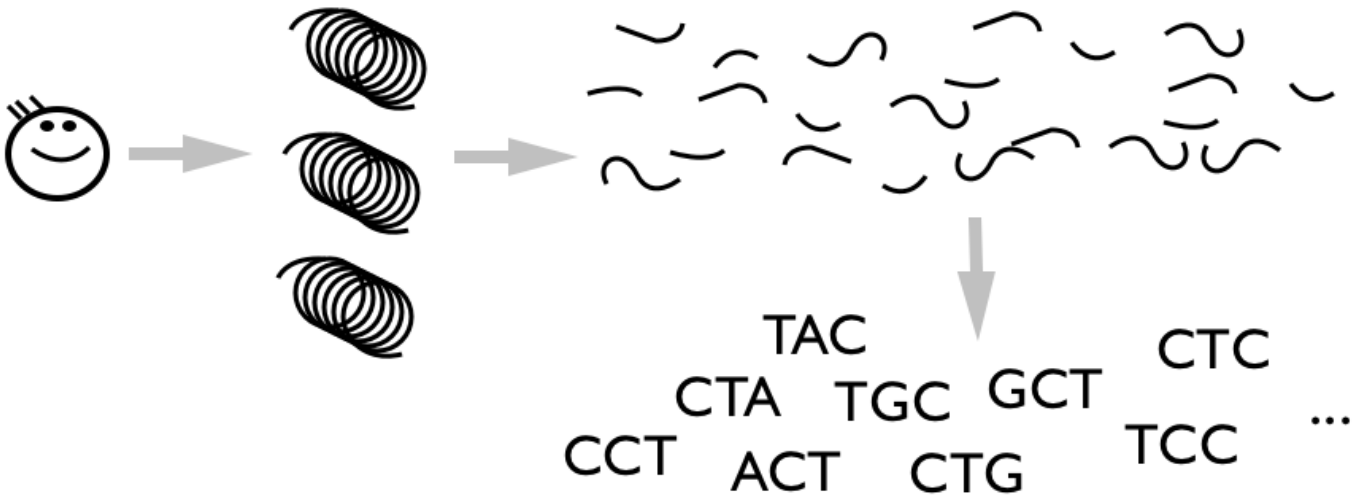
지식 생성 예(1): 염기서열 판독

"과감한" 컴퓨터 기술을 이용해서 비용문제 해결

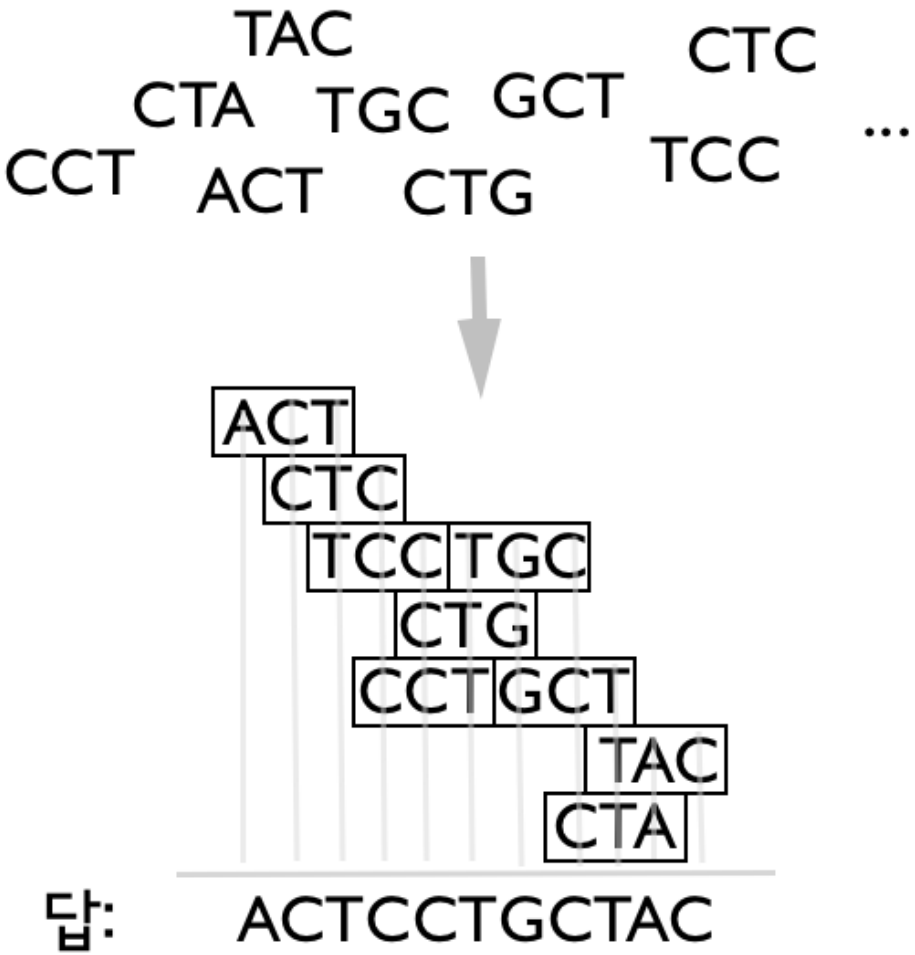
- 인간 유전자 염기서열(ACGT$\cdots$) 크기: 약 30억자 (3×10^9)
  - 약 100만 페이지 $=$ 1000 페이지 책 1000권
- 실험기자재로 한번에 판독할 수 있는 양: 100자이내
- 유전자 실 전체를 그 만큼씩 판독: 3×10^7 횟수필요.
- 비현실적인 시간과 비용.

구원투수(컴퓨터능력의 활용): 산탄총 방식(shotgun sequencing) 알고리즘

산탄총 방식(shotgun sequencing) 알고리즘(1/4)



산탄총 방식(shotgun sequencing) 알고리즘(2/4)



산탄총 방식(shotgun sequencing) 알고리즘(3/4)

양 끝의 염기서열들이 서로 겹치는 토막들을 연결해서
원래의 유전자 실을 복원하기

- 걸림돌: 초대량의 데이터
- 가능: 컴퓨터 기술 덕택

산탄총 방식(shotgun sequencing) 알고리즘(3/4)

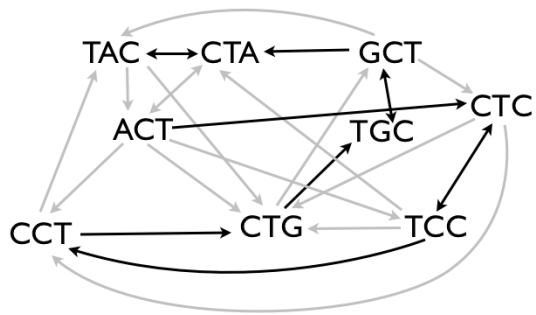
양 끝의 염기서열들이 서로 겹치는 토막들을 연결해서
원래의 유전자 실을 복원하기

- 걸림돌: 초대량의 데이터
- 가능: 컴퓨터 기술 덕택
- 문제:
  - 문자열 토막들 $\{s_0, s_1, \cdots, s_n\}$이 주어졌을 때, 모두를 품고 있는 하나의 가장 짧은 문자열 S 찾기.
  - 예)
    $\{ACT, CTA, TAC, CCT, CTG, TGC, TCC, GCT, CTC\}$
  - 답) $ACTCCTGCTAC$ 또는 $ACTGCTCCTAC$

산탄총 방식(shotgun sequencing) 알고리즘(4/4)

문제 모델

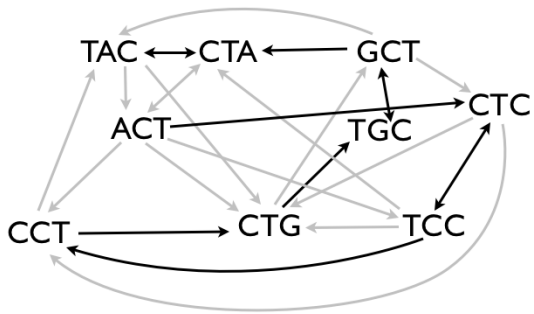
- 문자열 토막들의 끝이 겹치는 관계를 그래프로 표현하면, 그 그래프에서 [ ] 찾기.
- 입력:



산탄총 방식(shotgun sequencing) 알고리즘(4/4)

문제 모델

- 문자열 토막들의 끝이 겹치는 관계를 그래프로 표현하면, 그 그래프에서 [ ] 찾기.
- 입력:

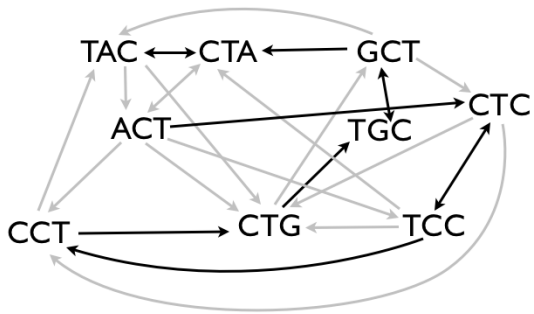


- 할일: 꼭지를 한번씩만 모두 방문하는 여정 찾기 (Hamiltonian path 찾기)

산탄총 방식(shotgun sequencing) 알고리즘(4/4)

문제 모델

- 문자열 토막들의 끝이 겹치는 관계를 그래프로 표현하면, 그 그래프에서 [ ] 찾기.
- 입력:

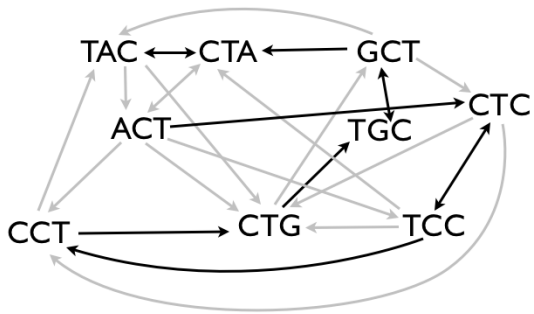


- 할일: 꼭지를 한번씩만 모두 방문하는 여정 찾기 (Hamiltonian path 찾기)
- "NP-complete" 문제

산탄총 방식(shotgun sequencing) 알고리즘(4/4)

문제 모델

- 문자열 토막들의 끝이 겹치는 관계를 그래프로 표현하면, 그 그래프에서 [ ] 찾기.
- 입력:

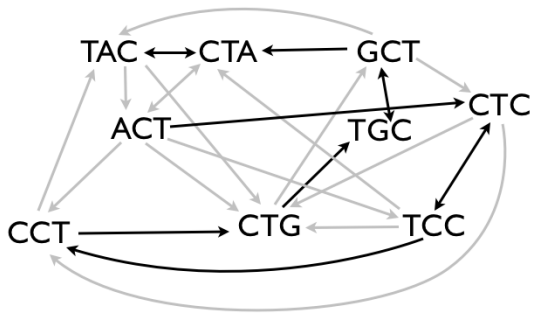


- 할일: 꼭지를 한번씩만 모두 방문하는 여정 찾기 (Hamiltonian path 찾기)
- "NP-complete" 문제
  - 이론: 운좋아야 현실적인 비용으로 가능

산탄총 방식(shotgun sequencing) 알고리즘(4/4)

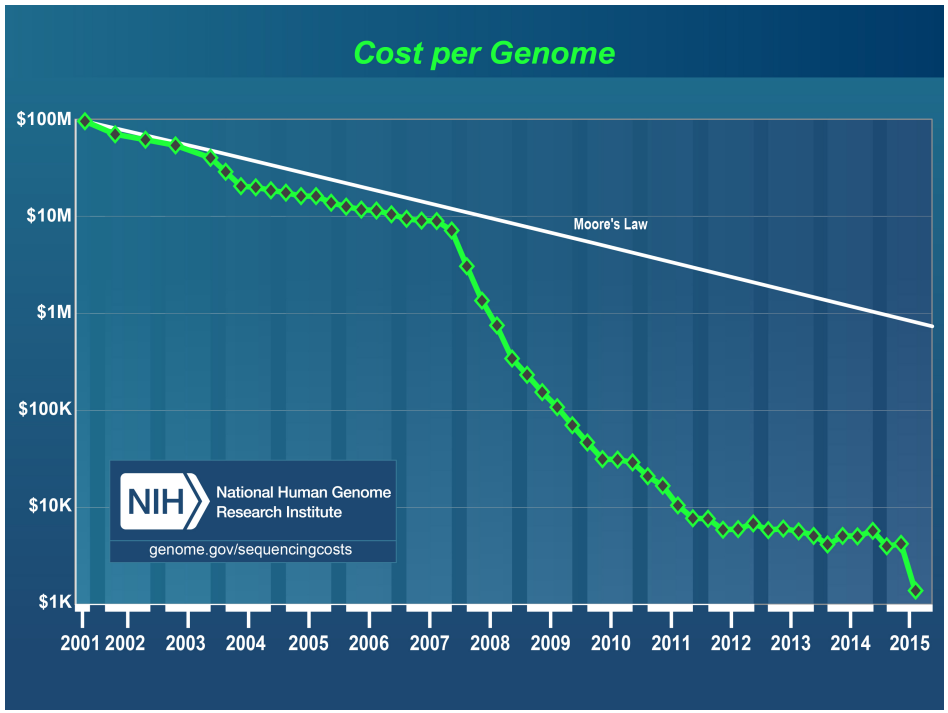
문제 모델

- 문자열 토막들의 끝이 겹치는 관계를 그래프로 표현하면, 그 그래프에서 [ ] 찾기.
- 입력:



- 할일: 꼭지를 한번씩만 모두 방문하는 여정 찾기 (Hamiltonian path 찾기)
- "NP-complete" 문제
  - 이론: 운좋아야 현실적인 비용으로 가능
  - 현실: 적절한 "통밥"(heuristics)을 이용하면 대개 현실적으로 가능

염기서열 판독비용 감소 속도

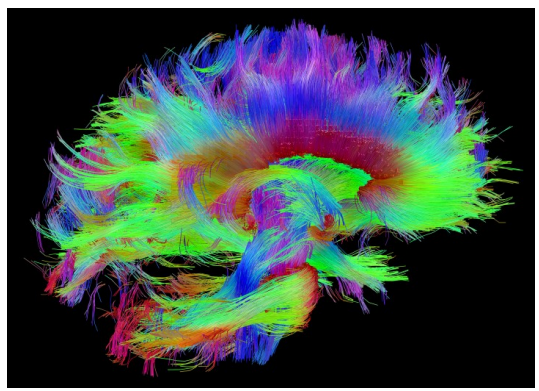


(사진출처: Google)

지식 생성 예(2): 인간 뇌 뉴런 구조 규명

Human Connectome Project, Wired Differently

- 1000억개의 뉴런/뇌, 접점 1만개/뉴련 = 500조 접점/뇌
- 인류가 지금까지 다뤄보지 못한 복잡도. 거대한 밀림



(사진출처: Google)

- 인간과 컴퓨터의 협업
- 컴퓨터 자동화 $\times$ 사람이 점검 보완
- 사람동원? 게임으로. "엄마, 인류를 위한 게임이예요!"

지식 생성 예(3): 거대한 인과관계 시스템 자동 파악

빅 메카니즘(Big Mechanism)

- 생명체 시스템, 대기권 날씨 시스템, 전지구적 경제현상
- 사람: 작은 인과관계의 파편들을 파악
- 컴퓨터: 쏟아지는 논문들을 종합해서 전체 인과관계 파악
- 쏟아지는 논문들의 양 $>>$ 전문가가 파악할 수 있는 양

현재 암 관련 논문들에 대해서

인간 지능의 확장 3: 지식 검색

컴퓨터없이는 불가능했던 검색

초광대한 지식의 탐색. 예전에는? 부분적인 지식의 탐색

- 손가락 끝에 제공되는 모든 지식들
- Google같은 탐색기술의 성과
  - 지식모음(collection): 이세상 모든 웹페이지, 책, 이미지
  - 인덱싱(indexing, tagging): 탐색준비
  - 랭킹(ranking): 탐색결과 정리

지능적인 지식 검색

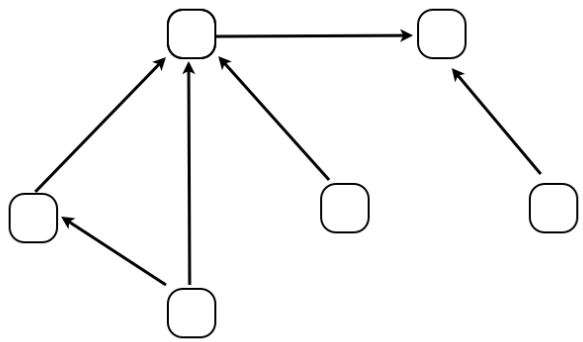
복수의 탐색 결과들 중에서 어느 것이 정답일까?

- 기준: 제일 많이 볼 자료(웹페이지, 책페이지, 이미지등)
  - 그런 웹페이지 순서대로 결과를 보여줘야
  - 그런 순서(질 순서)를 어떻게 알아내는가?

그런 순서를 컴퓨터 덕택에 유추할 수 있게 되었다.

제일 많이 보는 웹 페이지? 안1

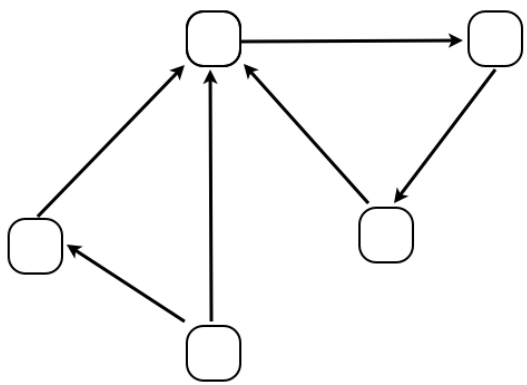
링크의 양으로: 많이 연결되어진 페이지 우선



꼭 제일 많이 보는 것은 아닌

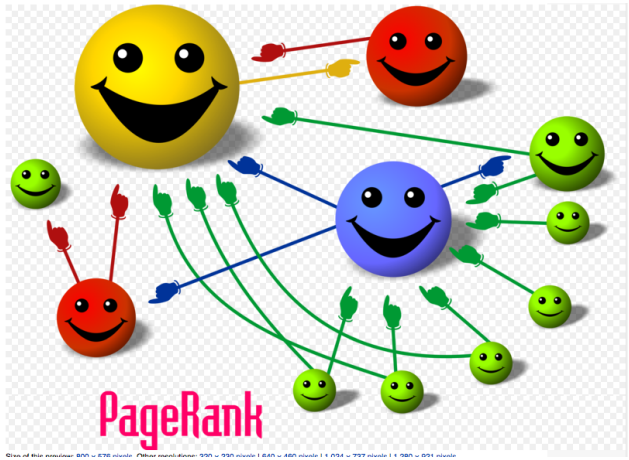
제일 많이 보는 웹 페이지? 안2

링크의 질로: 링크의 두께(많이 연결된 페이지가 연결하는 링크) 고려하기



물고무는 경우는 질이 ∞

제일 많이 보는 웹 페이지? Page & Brin



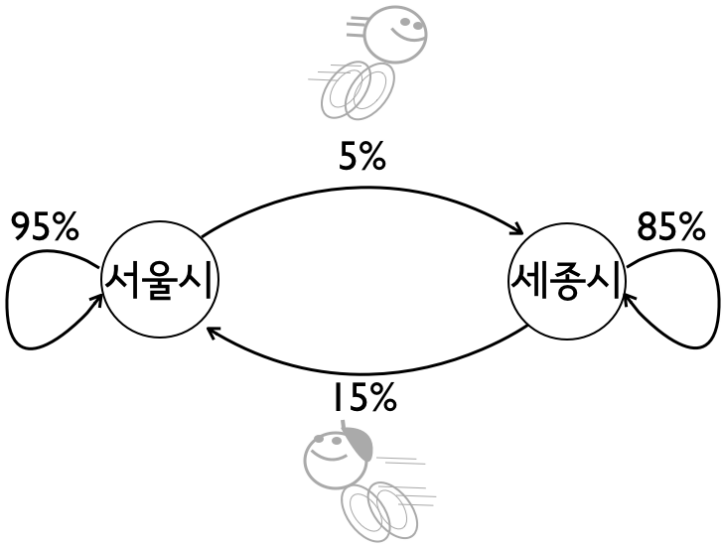
위의 문제없이 랭킹하는 방법:

- 페이지가 방문될 비율 계산
- 무대뽀: 무수히 많은 사람들의 웹서핑을 시뮬레이션하자
- 수리모델: Markov chain의 극한값(steady state)

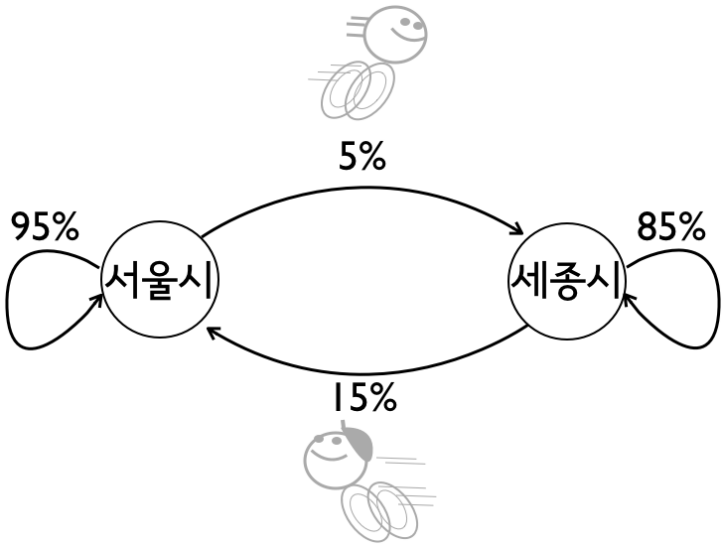
"사람들이 페이지를 방문하는 상대적인 비율을 예측하기"

(사진출처: Google)

마르코프 모델

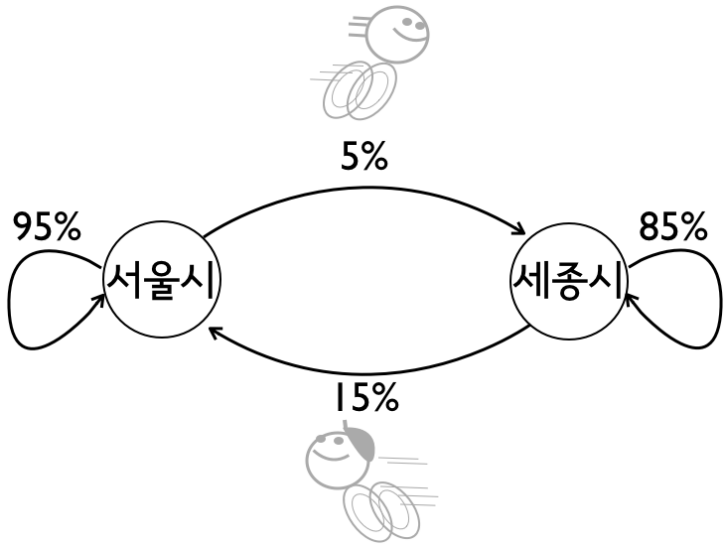


마르코프 모델



$$\text{내년 서울시 인구} = 0.95 \times \text{올해 서울시 인구} + 0.15 \times \text{올해 세종시 인구}$$
$$\text{내년 세종시 인구} = 0.05 \times \text{올해 서울시 인구} + 0.85 \times \text{올해 세종시 인구}$$

마르코프 모델

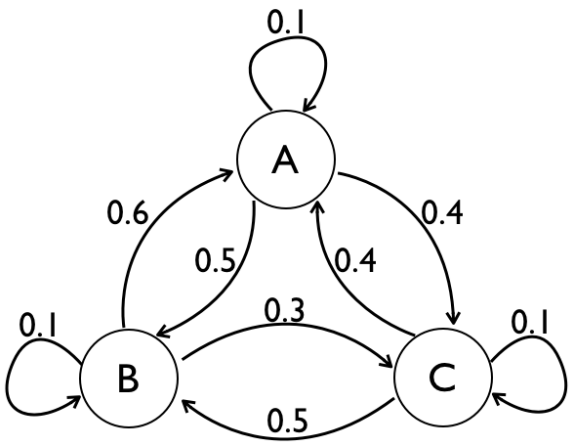


내년 서울시 인구 $=$ $0.95 \times$ 올해 서울시 인구 $+ 0.15 \times$ 올해 세종시 인구

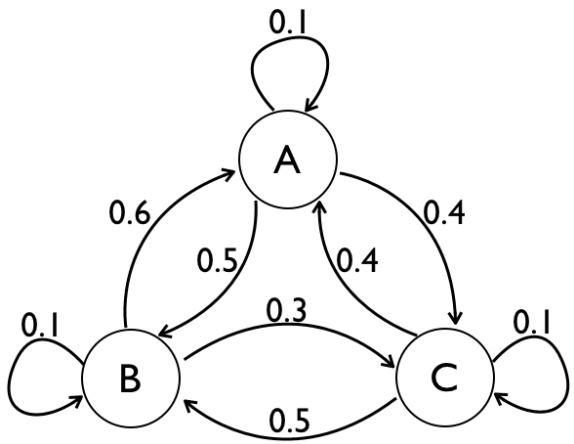
내년 세종시 인구 $=$ $0.05 \times$ 올해 서울시 인구 $+ 0.85 \times$ 올해 세종시 인구

	올해	내년	내후년	$\cdots$	언젠가는
서울시 인구	10.00	9.65	9.37	$\cdots$	8.25
세종시 인구	1.00	1.35	1.63	$\cdots$	2.75

마르코프 모델

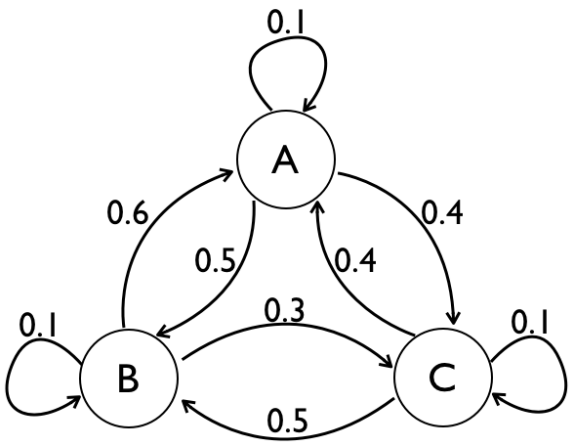


마르코프 모델



$$
\begin{aligned}
A_{k+1} &= 0.1 \times A_k + 0.6 \times B_k + 0.4 \times C_k \\
B_{k+1} &= 0.5 \times A_k + 0.1 \times B_k + 0.5 \times C_k \\
C_{k+1} &= 0.4 \times A_k + 0.3 \times B_k + 0.1 \times C_k
\end{aligned}
$$

마르코프 모델



$$
\begin{aligned}
A_{k+1} &= 0.1 \times A_k + 0.6 \times B_k + 0.4 \times C_k \\
B_{k+1} &= 0.5 \times A_k + 0.1 \times B_k + 0.5 \times C_k \\
C_{k+1} &= 0.4 \times A_k + 0.3 \times B_k + 0.1 \times C_k
\end{aligned}
$$

	$k=0$	$k=1$	$k=2$	$\cdots$	$k=\infty$
A_k	$1/3$	0.367	0.363	$\cdots$	0.363
B_k	$1/3$	0.367	0.353	$\cdots$	0.357
C_k	$1/3$	0.266	0.284	$\cdots$	0.280

Google's PageRank 알고리즘

수리모델:

- Markov matrix $\mathcal{M}$
  - x_{ij}: i-페이지가 j-페이지를 링크하면 $1/n_i$
    (n_i = i-페이지에서 외부로 나가는 링크 수)
- 목표: $S = \lim_{i\to\infty} S_i$, 여기서 $S_{k+1} = \mathcal{M}(S_k)$
- 실제: $S = \lim_{i\to\infty} S_i$, 여기서 $S_{k+1} = \mathcal{N}(S_k)$ 이고
  $\mathcal{N} = \mathcal{M} \oplus \epsilon$(0이 없는 Markov 행렬)
  - 왜? Markov 행렬안의 모두가 양이면, 그 Markov chain의 극한값은 유일하게 존재. (Perron-Frobenius 정리)

다른 기준들도: 예) 질의문 단어들의 폰트크기, 웹페이지 단어들과의 연관도

지식 검색 시설

- 옛날 스타일:

- 요즘 스타일: 네이버 데이터센터, Google Server Farm



(사진출처: Google)

인간 지능의 확장 4: 팀웍 지능

사람과 컴퓨터의 협업

인간 지능의 확장 4: 팀웍 지능

사람과 컴퓨터의 협업

- 사람은 쉬운 일 & 컴퓨터는 어려운 일: 사람에게 시키기

인간 지능의 확장 4: 팀웍 지능

사람과 컴퓨터의 협업

- 사람은 쉬운 일 & 컴퓨터는 어려운 일: 사람에게 시키기
- 할일이 많다? 많은 사람을 동원

인간 지능의 확장 4: 팀웍 지능

사람과 컴퓨터의 협업

- 사람은 쉬운 일 & 컴퓨터는 어려운 일: 사람에게 시키기
- 할일이 많다? 많은 사람을 동원
- 어떤 유인책? 재미로(게임), 의미로(사회운동), 득으로(공부)

인간 지능의 확장 4: 팀웍 지능

사람과 컴퓨터의 협업

- 사람은 쉬운 일 & 컴퓨터는 어려운 일: 사람에게 시키기
- 할일이 많다? 많은 사람을 동원
- 어떤 유인책? 재미로(게임), 의미로(사회운동), 득으로(공부)
- “human computation”, “Luis von Ahn”

인간 지능의 확장 4: 팀웍 지능

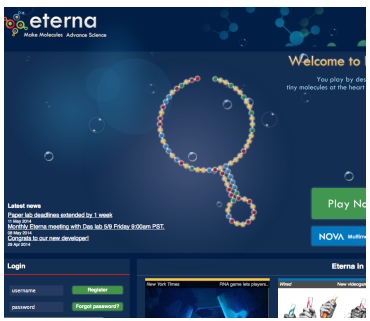
사람과 컴퓨터의 협업

- 사람은 쉬운 일 & 컴퓨터는 어려운 일: 사람에게 시키기
- 할일이 많다? 많은 사람을 동원
- 어떤 유인책? 재미로(게임), 의미로(사회운동), 득으로(공부)
- “human computation”, “Luis von Ahn”
- ESP Game, reCHAPCHA, Duolingo 등등

인간 지능의 확장 4: 군중 지능

컴퓨터가 모으는 대규모 세계인

- 컴퓨터덕택에 같은 뜻을 가진 누구나 쉽게 모인다
- 군중에 맡겨 공통의 문제를 푼다(예. FoldIt, EteRNA, Eyewire)



(사진출처: Google)

$$\text{일반 군중} \times \text{컴퓨터게임} > \text{전문가}$$
$$\text{일반 군중} \times \text{컴퓨터게임} > \text{컴퓨터}$$

- 시민과학(citizen science)

컴퓨터과학의 응용 II: 우리 본능의 확장

컴퓨터가 응원하고 확대시켜주는 인간의 본능

- 놀고싶은 본능
- 소통의 본능

우리 본능의 확장 1: 놀고싶은 본능

- 컴퓨터 게임: 컴퓨터와, 친구와, 불특정 다수의 사람들과
- MMORPG(massively multiplayer online role-playing game)
- "에버랜드"보다 거대한 가상세계들이 컴퓨터안에

우리 본능의 확장 1: 놀고싶은 본능

- 컴퓨터 게임: 컴퓨터와, 친구와, 불특정 다수의 사람들과
- MMORPG(massively multiplayer online role-playing game)
- "에버랜드"보다 거대한 가상세계들이 컴퓨터안에
- 가능성과 순기능
  - *Reality Is Broken: why games make us better and how they can change the world*, Jane McGonigal
  - *한국형 디지털 스토리텔링: [리니지2] 바츠 해방 전쟁 이야기*, 이인화 (특히 4부)

우리 본능의 확장 2: 소통의 본능

이메일, 카카오톡, facebook, twitter, google hangout, skype,

음악/동영상 공유 torrent, melon, bugs

- 놀라운 첫번째 화살: 속도와 스케일
- 더 놀라운 두번째 화살: 온전히 전달된다

근간 기술

- 정보이론(information theory)
- 온전히 전달하는 방법? 코딩(encoding + error-correcting)기술

정보이론 이전까지는

통신이 무엇인지에 대한 이해 부족

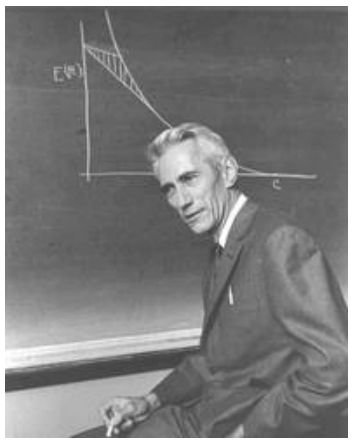
- 주먹구구식: 라디오, 전보, 텔레비젼등 경우마다 그럭저럭

통신잡음 극복법?

- 통신중 잡음/에러: 메세지 손상
- 잡음은 있게마련
- 잡음은 통신의 물리적인 현상
- 물리적으로 해결할 수 밖에: 큰 소리로?
- 좌절: 잡음도 같이 커지네! 다른 방법이 없는 듯

정보이론(information theory)

통신이란 무엇인가에 대한 답, 디지털 소통의 근간기술



Claude Shannon(1916 – 2001)

- "A Mathematical Theory of Communication", Bell System Technical Journal 27(3):379-423, 27(4):623-656, 1948, Bell Labs
- "Margna Carta of information age"

(사진출처: Google)

디지털 소통의 근간기술: 정보이론 (1/4)

혁신/복음: 통신의 한계는

디지털 소통의 근간기술: 정보이론 (1/4)

혁신/복음: 통신의 한계는

- 물리가(통신채널) 아니라 메세지가 가진 정보량이다

디지털 소통의 근간기술: 정보이론 (1/4)

혁신/복음: 통신의 한계는

- 물리가(통신채널) 아니라 메세지가 가진 정보량이다
- 잡음은 어쩔 수 없지만 극복방법은 존재한다

디지털 소통의 근간기술: 정보이론 (1/4)

혁신/복음: 통신의 한계는

- 물리가(통신채널) 아니라 메세지가 가진 정보량이다
- 잡음은 어쩔 수 없지만 극복방법은 존재한다
- 그 방법은 하드웨어(물리채널)에 있지않고 소프트웨어(메세지 정보량)에 있다

디지털 소통의 근간기술: 정보이론 (1/4)

혁신/복음: 통신의 한계는

- 물리가(통신채널) 아니라 메세지가 가진 정보량이다
- 잡음은 어쩔 수 없지만 극복방법은 존재한다
- 그 방법은 하드웨어(물리채널)에 있지않고 소프트웨어(메세지 정보량)에 있다

통신의 주인공은 메세지가 가진 정보량이다!

디지털 소통의 근간기술: 정보이론 (2/4)

메세지의 정보량?

- 메세지 겉모습에 대한 량
- 정보량? 잦은 것은 정보량이 적고 드문 것은 많다
- 정보량? 엔트로피: 글자의 예측불허의 정도
- 정보량? 메세지에 불필요한 글자가 끼면 정보량은 준다

디지털 소통의 근간기술: 정보이론 (3/4)

- 정의: 메세지의 정보량

$$H = -\sum_{x} p(x) \log_2 p(x).$$

디지털 소통의 근간기술: 정보이론 (3/4)

- 정의: 메세지의 정보량

$$H = -\sum_{x} p(x) \log_2 p(x).$$

- 정리1(채널 잡음이 없는 경우)

  채널의 용량이 C이고 정보량이 H이면
  메세지는 초당 최대 C/H로 항상 가능하다.

디지털 소통의 근간기술: 정보이론 (3/4)

- 정의: 메세지의 정보량

$$H = -\sum_{x} p(x) \log_2 p(x).$$

- 정리1(채널 잡음이 없는 경우)

  채널의 용량이 C이고 정보량이 H이면
  메세지는 초당 최대 C/H로 항상 가능하다.

  - 통신의 한계는 메세지의 정보량에 있다!

디지털 소통의 근간기술: 정보이론 (3/4)

- 정의: 메세지의 정보량

$$H = -\sum_{x} p(x) \log_2 p(x).$$

- 정리1(채널 잡음이 없는 경우)

  채널의 용량이 C이고 정보량이 H이면
  메세지는 초당 최대 C/H로 항상 가능하다.

  - 통신의 한계는 메세지의 정보량에 있다!

- 정리2(채널 잡음이 있는 경우)

  $H \leq C$이면, 온전히 전달 가능하다. $H > C$
  이면, 온전히 전달 불가능하고 오류율을 $H - C$
  이하로 줄일 수 없다.

디지털 소통의 근간기술: 정보이론 (3/4)

- 정의: 메세지의 정보량

$$H = -\sum_{x} p(x) \log_2 p(x).$$

- 정리1(채널 잡음이 없는 경우)

  채널의 용량이 C이고 정보량이 H이면
  메세지는 초당 최대 C/H로 항상 가능하다.

  - 통신의 한계는 메세지의 정보량에 있다!

- 정리2(채널 잡음이 있는 경우)

  $H \leq C$이면, 온전히 전달 가능하다. $H > C$
  이면, 온전히 전달 불가능하고 오류율을 $H - C$
  이하로 줄일 수 없다.

  - 아무리 잡음이 있어도 온전히 전달가능하다!

디지털 소통의 근간기술: 정보이론 (4/4)

- 메세지를 온전히 보낼 방법들?
  - 정보량이 채널용량보다 적게해서 보내면 가능하다
  - 예) 반복: “밥먹자” vs “밥먹자 밥먹자 밥먹자”
  - 효율적인 다양한 방법들 가능
- 정보이론(information theory)으로 분석가능
  - 그런 방법들의 오류율이 충분히 작은가? ($< 1/10^{25}$?)
  - 더 좋은 방법이 가능한가?

메세지 전달하기: 인코딩(encoding)× 오류수정장치(error-correcting)

인코딩(encoding)

예) 사용단어: “가마”, “꼭”, “꽃”, “타고”

문장예) “가마가마꼭가마꽃가마타고가마”, “꽃가마꼭타고가마가마꼭가마”

- 크기고정코드:
  - 00(가마), 01(타고), 10(꼭), 11(꽃). 복구?
- 크기변동코드: 단어빈도(가마 $>$ 타고 $>$ 꼭 $>$ 꽃) 이용
  - 0(가마), 11(타고), 100(꼭), 101(꽃). 복구?

메세지 전달하기: 인코딩(encoding)× 오류수정장치(error-correcting)

오류수정장치(error-correcting)

- 반복(repetition): “밥먹자?” ⇒ “밥먹자? 밥먹자? 밥먹자?”
  - 복구: “밥먹자? 밤먹자? 밥먹저?” ⇒ “밥먹자”
- 여분(redundancy): “027013” ⇒ “zero two seven zero one three”
  - 복구: “zeri dwo sevem zirp ome tjree” ⇒ “027013”

메세지 전달하기: 인코딩(encoding)× 오류수정장치(error-correcting)

오류수정장치(error-correcting)

- 검산치(checksum): “4 6 7” $\Rightarrow$ “4 6 7 7”
  - 복구: “4 6 7 5” $\Rightarrow$ “다시보내줘”(검산치 에러)
- 족집게 검산치(pinpoint checksum): “4 6 7 2 3 8 8 2 6”

  $\Rightarrow$

$$\begin{array}{cccc} 4 & 6 & 7 & 7 \\ 2 & 3 & 8 & 3 \\ 8 & 2 & 6 & 6 \\ 4 & 1 & 1 & \end{array}$$

컴퓨터과학의 응용 III: 우리 현실의 확장

컴퓨터가 확장시켜주는 현실: 전세계 누구와도 시공간 공유

- (예) 전세계 누구와도 가위바위보
  - 이전: "여기 현재" 있는 사람들 끼리만
- 전세계 불특정다수와 상거래
  - 이전: "동네" 사람들 끼리만
- 근간 기술: 컴퓨터 한계를 역이용
  - 암호기술(cryptography)

우리 현실의 확장

전세계 누구와도 가위바위보/상거래를 성사시키는 기술

- 컴퓨터로 간단한 그러나 그 역은 "불가능"한 계산 이용
- 역발상: NP 문제를 역이용하자

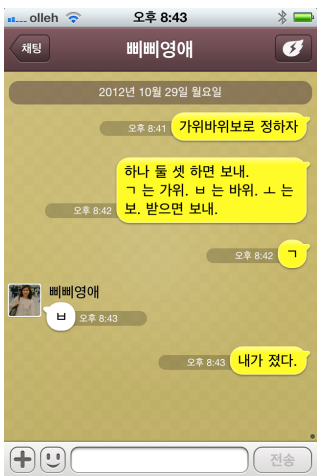
곱하기 vs 역: 소인수 분해, 현실적으로 불가능

P로 나눈 나머지 역: 현실적으로 불가능

전세계 누구와도 시공간공유: 가위바위보

시공간 공유해야: 동시에 내고 그자리에서 확인할 수 있어야

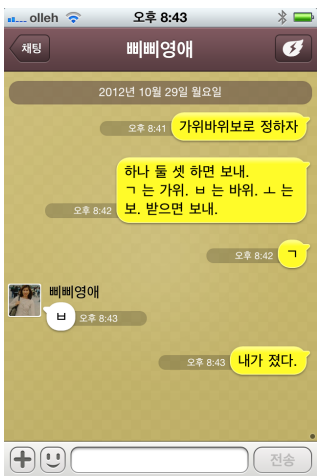
- 카카오톡, facebook, skype에서 가위바위보?
- 떨어져있는 전세계 친구들과?



전세계 누구와도 시공간공유: 가위바위보

시공간 공유해야: 동시에 내고 그자리에서 확인할 수 있어야

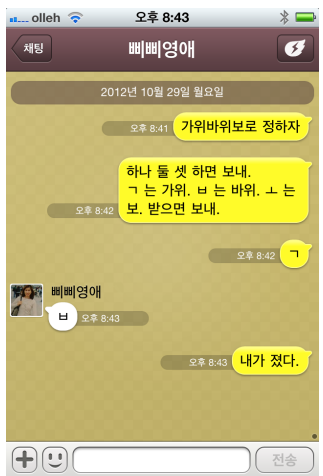
- 카카오톡, facebook, skype에서 가위바위보?
- 떨어져있는 전세계 친구들과?
  - 문제: 속일 수 있는 여지
  - 늦게내기: 온 것 보고 보내기



전세계 누구와도 시공간공유: 가위바위보

시공간 공유해야: 동시에 내고 그자리에서 확인할 수 있어야

- 카카오톡, facebook, skype에서 가위바위보?
- 떨어져있는 전세계 친구들과?
  - 문제: 속일 수 있는 여지
  - 늦게내기: 온 것 보고 보내기
  - 심판을 두기? 심판이 거짓말하면?



전세계 누구와도 시공간공유: 가위바위보

컴퓨터로 해결하는 방법

- 속일 수 없게. 믿고 맡기는 심판도 없이

전세계 누구와도 시공간공유: 가위바위보

컴퓨터로 해결하는 방법

- 속일 수 없게. 믿고 맡기는 심판도 없이
- 소인수 분해의 어려움을 이용

전세계 누구와도 시공간공유: 가위바위보

컴퓨터로 해결하는 방법

- 속일 수 없게. 믿고 맡기는 심판도 없이
- 소인수 분해의 어려움을 이용
  - 가위/바위/보 마다 1/3/7로 끝나는 솟수를 사용하기로 하고

전세계 누구와도 시공간공유: 가위바위보

컴퓨터로 해결하는 방법

- 속일 수 없게. 믿고 맡기는 심판도 없이
- 소인수 분해의 어려움을 이용
  - 가위/바위/보 마다 1/3/7로 끝나는 솟수를 사용하기로 하고
  - 각자 낼 것에 해당하는 소수와 그보다 작은 소수를 곱해서 보냄(X, Y)

전세계 누구와도 시공간공유: 가위바위보

컴퓨터로 해결하는 방법

- 속일 수 없게. 믿고 맡기는 심판도 없이
- 소인수 분해의 어려움을 이용
  - 가위/바위/보 마다 1/3/7로 끝나는 솟수를 사용하기로 하고
  - 각자 낼 것에 해당하는 소수와 그보다 작은 소수를 곱해서 보냄(X, Y)
  - 서로 받고 나면 소인수분해 못함(서로가 낸 것을 못봄)

전세계 누구와도 시공간공유: 가위바위보

컴퓨터로 해결하는 방법

- 속일 수 없게. 믿고 맡기는 심판도 없이
- 소인수 분해의 어려움을 이용
  - 가위/바위/보 마다 1/3/7로 끝나는 솟수를 사용하기로 하고
  - 각자 낼 것에 해당하는 소수와 그보다 작은 소수를 곱해서 보냄(X, Y)
  - 서로 받고 나면 소인수분해 못함(서로가 낸 것을 못봄)
  - 서로에게 자기 소수를 공개(소인수 x, y)

전세계 누구와도 시공간공유: 가위바위보

컴퓨터로 해결하는 방법

- 속일 수 없게. 믿고 맡기는 심판도 없이
- 소인수 분해의 어려움을 이용
  - 가위/바위/보 마다 1/3/7로 끝나는 솟수를 사용하기로 하고
  - 각자 낼 것에 해당하는 소수와 그보다 작은 소수를 곱해서 보냄(X, Y)
  - 서로 받고 나면 소인수분해 못함(서로가 낸 것을 못봄)
  - 서로에게 자기 소수를 공개(소인수 x, y)
  - 서로 받은 숫자를 받은 소수로 나눠봄(X/x, Y/y); 서로 승패를 확인

전세계 누구와도 시공간공유: 가위바위보

컴퓨터로 해결하는 방법

- 속일 수 없게. 믿고 맡기는 심판도 없이
- 소인수 분해의 어려움을 이용
  - 가위/바위/보 마다 1/3/7로 끝나는 솟수를 사용하기로 하고
  - 각자 낼 것에 해당하는 소수와 그보다 작은 소수를 곱해서 보냄(X, Y)
  - 서로 받고 나면 소인수분해 못함(서로가 낸 것을 못봄)
  - 서로에게 자기 소수를 공개(소인수 x, y)
  - 서로 받은 숫자를 받은 소수로 나눠봄(X/x, Y/y); 서로 승패를 확인
  - (세 소수를 곱해서보내면? 나눈 값 X/x가 소수인지 확인은 쉽다)

암호기술(cryptography) I: 기밀 통신

두 사람이 미리 비밀 "키"(key)를 공유

- 메세지 보내는 사람과 받는 사람만 알 수 있게
  - 암호걸기: 보내는 사람은 메세지 글자들을 비밀키로 "더해서" 보내기
  - 암호풀기: 받는 사람은 메세지 글자들을 비밀키로 "빼서" 암호를 풀기
- 단순히 "더하고 빼기"는 아니지만

암호기술(cryptography)

비밀키를 공유해야 하는 문제점

- 누구도 모르게, 미리 서로 정해놔야. 당사자와 미리 만나야!

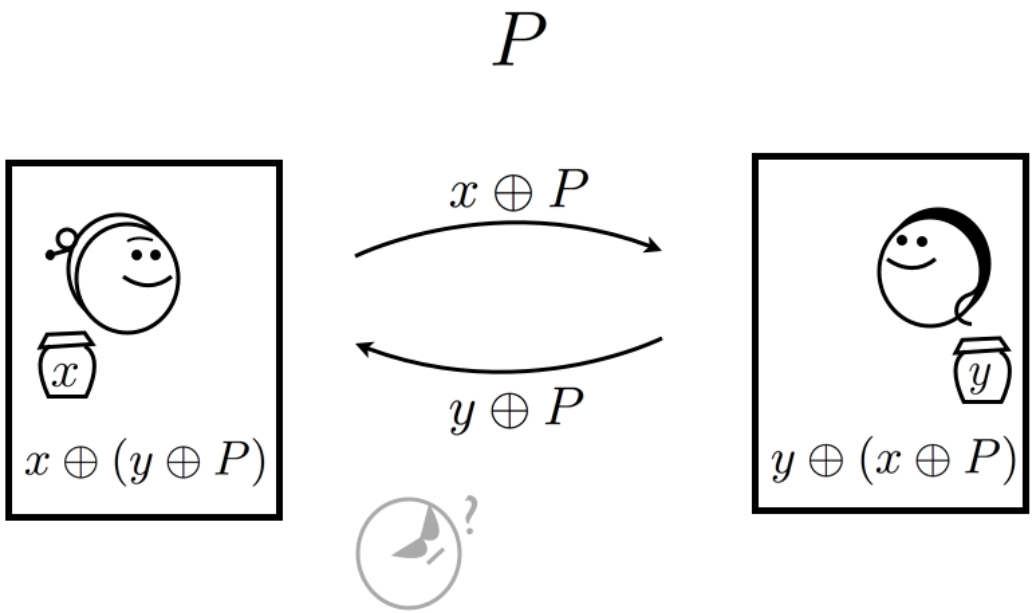
암호기술(cryptography)

비밀키를 공유해야 하는 문제점

- 누구도 모르게, 미리 서로 정해놔야. 당사자와 미리 만나야!
- 불특정 다수와 거래하고 싶은데?
- 비밀소통(거래) 하려면 비밀소통(비밀키)가 미리 필요. 닭 vs 달걀

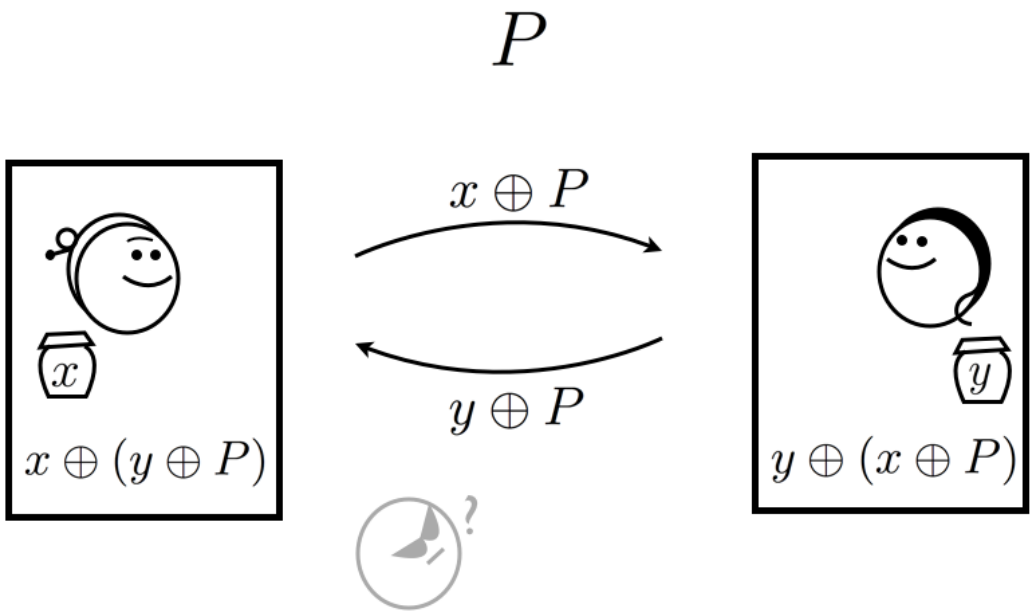
공개키 암호기술(public key cryptography)

비밀키를 공개적으로 만들 수 있는 방법? “Diffie-Hellman key exchange”



공개키 암호기술(public key cryptography)

비밀키를 공개적으로 만들 수 있는 방법? “Diffie-Hellman key exchange”



$$x \oplus P = 2^x \bmod P \qquad y \oplus P = 2^y \bmod P$$

$$y \oplus (x \oplus P) = (x \oplus P)^y \bmod P \qquad x \oplus (y \oplus P) = (y \oplus P)^x \bmod P$$

공개키 암호기술(public key cryptography)

`https:`로 열리는 페이지에서 주고받는 정보들은 이 방식으로

- 예) amazon.com에서 책을 살때: 신용카드 정보
- "128-bit encryption": 비밀키의 크기가 128 bits

암호기술 II: 완벽한 하인

기밀 통신 다음으로

- 암호화한 데이터 $\underline{x}$를 가지고 일을 해야한다면
- 암호풀고; 일 마치고; 다시 암호걸고?
  - 위험: 일하는 사람을 믿어야

암호기술 II: 완벽한 하인

기밀 통신 다음으로

- 암호화한 데이터 x를 가지고 일을 해야한다면
- 암호풀고; 일 마치고; 다시 암호걸고?
  - 위험: 일하는 사람을 믿어야
- 암호풀지 않고 가능할까?

암호기술 II: 완벽한 하인

기밀 통신 다음으로

- 암호화한 데이터 $\underline{x}$를 가지고 일을 해야한다면
- 암호풀고; 일 마치고; 다시 암호걸고?
  - 위험: 일하는 사람을 믿어야
- 암호풀지 않고 가능할까?
  - 동형암호(homomorphic encryption) (2009년)
  - 모든 sw f에 대해서 f'이 존재:

$$\underline{f(x)} = f'(\underline{x})$$

암호기술 II: 완벽한 하인

기밀 통신 다음으로

- 암호화한 데이터 $\underline{x}$를 가지고 일을 해야한다면
- 암호풀고; 일 마치고; 다시 암호걸고?
  - 위험: 일하는 사람을 믿어야
- 암호풀지 않고 가능할까?
  - 동형암호(homomorphic encryption) (2009년)
  - 모든 sw f에 대해서 f'이 존재:

$$\underline{f(x)} = f'(\underline{x})$$

  - 예) $\underline{n} = n + p \times a$ (열쇠 p). 그러면 $+' = +$.

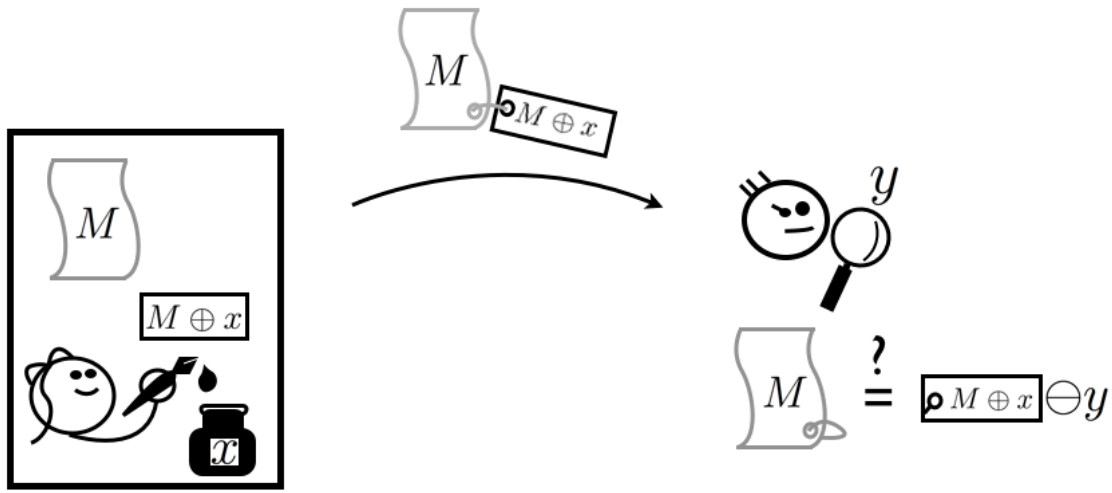
암호기술 III: 진품감정

온전히 소통하며 시공간 뛰어 넘기

- 잡음 극복: 정보이론
- 엿듣기 극복: 암호기술 (공개키, 동형암호)
- 짝퉁 극복: 암호기술 (진품감정)

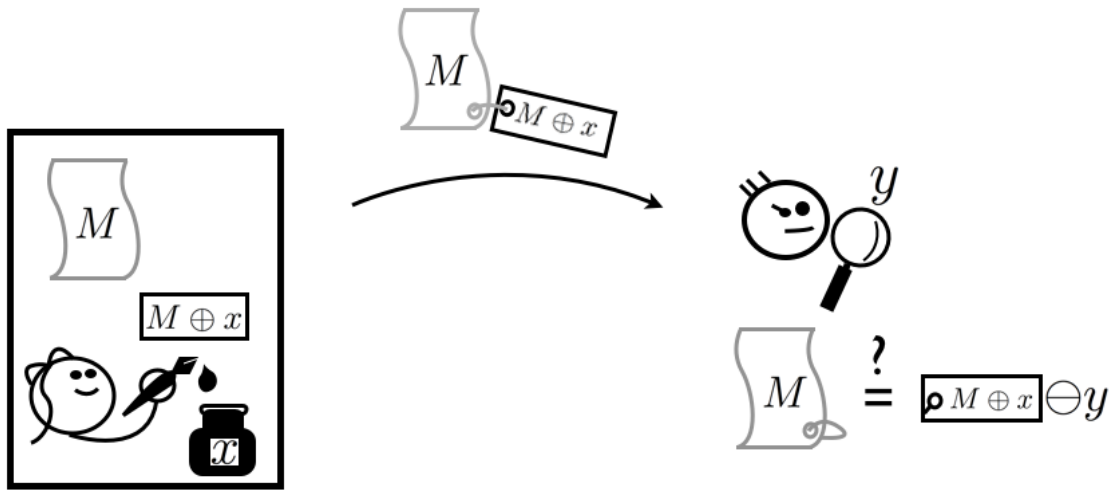
진품감정

- 그녀의 열쇠짝 (x, y), 그녀의 작품 M
- 서명하는 손 x (비밀): $M \oplus x$
- 감정하는 눈 y (공개): $(M \oplus x) \ominus y = M$?



진품감정

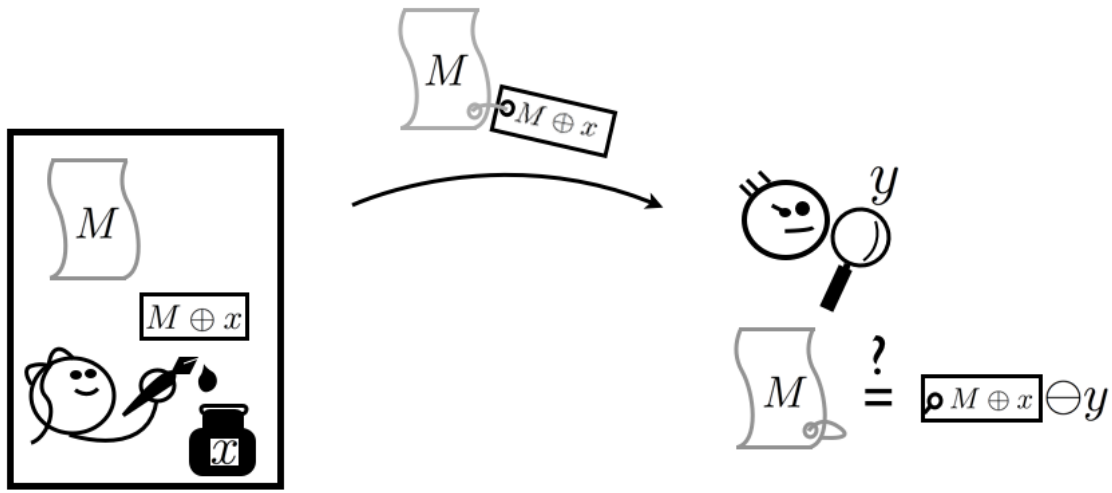
- 그녀의 열쇠짝 (x, y), 그녀의 작품 M
- 서명하는 손 x (비밀): $M \oplus x$
- 감정하는 눈 y (공개): $(M \oplus x) \ominus y = M$?



- 1977년: 실용적인 $\oplus, \ominus$가 찾아짐 (Rivest, Shmir, Adleman)

진품감정

- 그녀의 열쇠짝 (x, y), 그녀의 작품 M
- 서명하는 손 x (비밀): $M \oplus x$
- 감정하는 눈 y (공개): $(M \oplus x) \ominus y = M$?



- 1977년: 실용적인 $\oplus, \ominus$가 찾아짐 (Rivest, Shmir, Adleman)
- 응용: 서명하는 손(공개), 감정하는 눈(비밀) $\Longrightarrow$ 불특정다수가 나만 읽을 수 있는 비밀메세지 전달

벼랑

디지털 컴퓨터의 한계에 기대고 있는 현재의 암호기술

- 디지털 컴퓨터로 인수분해를 빨리하는 알고리즘은 아직 못찾음
  - 인수분해는 $\mathcal{NP}$문제
  - $\mathcal{P} \neq \mathcal{NP}$는 아직 모름
- 퀀텀 컴퓨터로는 인수분해를 빨리하는 알고리즘이 존재
  - 다른 문제를 찾아야: 수학에는 많다

벼랑

디지털 컴퓨터의 한계에 기대고 있는 현재의 암호기술

- 디지털 컴퓨터로 인수분해를 빨리하는 알고리즘은 아직 못찾음
  - 인수분해는 $\mathcal{NP}$문제
  - $\mathcal{P} \neq \mathcal{NP}$는 아직 모름
- 퀀텀 컴퓨터로는 인수분해를 빨리하는 알고리즘이 존재
  - 다른 문제를 찾아야: 수학에는 많다

아이러니

- 대책없는 문제들이 유용하게 쓰인다니
- 수백년간 인류에게 좌절만 안겨준 수학 문제들
- 인류의 시공간을 확장하는 데 유용

블록체인(blockchain) 알고리즘 (1/2)

믿고맡길 중간자없이
여럿이 만장일치 자료를
유지관리하도록 해주는 알고리즘

블록체인(blockchain) 알고리즘 (1/2)

믿고맡길 중간자없이
여럿이 만장일치 자료를
유지관리하도록 해주는 알고리즘

- "여럿이": 흩어져있는 컴퓨터들이
- "만장일치 자료를 유지관리하도록": 모두가 같은 자료를 똑같이 갱신하도록
- "해주는 알고리즘": 모두가 같은 알고리즘으로

블록체인(blockchain) 알고리즘 (2/2)

믿고맡길 중간자없이
여럿이 만장일치 자료를
유지관리하도록 해주는 알고리즘

- "믿고맡길 중간자없이"
  - 불필요: 중앙정부, 중앙은행, 관공서, 대학본부 등
- 그런 자료?
  - 모두가 믿을 자료
  - 조작 불가능해야하는 자료

블록체인의 응용들

블록체인의 응용들

- 모든 가치있는 것들의 거래/판매 이력을 블록체인에 기록
  - 돈/부동산/예술품/노래/책 등이
  - 얼마만큼 누구에게 이동했는지 시간순으로

블록체인의 응용들

- 모든 가치있는 것들의 거래/판매 이력을 블록체인에 기록
  - 돈/부동산/예술품/노래/책 등이
  - 얼마만큼 누구에게 이동했는지 시간순으로
- 조작불가능해야 할 데이터 보관용
  - 투표용지
  - 본인확인 돋보기(public key) 표: 이름-돋보기 쌍들
    1. 내 싸인키(private key)로 "나는 이광근이다. 농협계좌이체하려고 한다."
    2. 농협에서는 블록체인에 있는(믿을만한) 이광근 돋보기로 확인.
    - (생체인식 방법등도 있지만)

블록체인 알고리즘 (1/4)

믿고맡길 중간자없이
여럿이 만장일치 자료를
유지관리하도록 해주는 알고리즘

- 예: 모두의 돈 이동 내역
- 상황: 두 군데서 거래 발생
  - (A가, B에게, 100원) 와 (A가, C에게, 500원)
- 목표
  - 모두가 만장일치로 기록해야
  - 기록된 것은 누구도 고치지 못해야

블록체인 알고리즘 (2/4)

모두가 만장일치로 기록해야

- 누구나 자기 장부에 그냥 쓸 수 있으면 만장일치 유지 불가
  - A 잔고가 510원이라면
  - 누구는 (A가, B에게, 100원)만 기록가능.
  - 누구는 (A가, B에게, 500원)만 기록가능.

알고리즘

- 기록할때마다 대표를 선발
- 선발방식: 어떤 능력을 보인 자를 대표로
- 만장일치: 대표 (능력으로 검증한) 블락을 모두가 복사
- 동기부여: 대표 역할을 하면 보상

블록체인 알고리즘 (3/4)

대표 선발방식: 어떤 능력을 보여야 대표

- 방식-I. 이만큼 해냈어(POW, proof of work)
  - 누군가 이만큼 할 수 있다고 보임.
  - "그 정도 일을 해냈"으므로 모두가 대표로 인정.
- 방식-II. 이만큼 걸었어(POS, proof of stake)
  - 각자 돈을 검. 돈비례로 대표+검증단 무작위 선발.
  - "돈비례 무작위"이므로 모두가 대표+검증단 인정.

블록체인 알고리즘: Proof of Work

1. 새 거래가 뜸
2. 모두 대표가 되기위해 경쟁: 블록(거래내역)을 매달 체인을 찾는다
3. 체인을 찾은 자가 현재 블록을 매달고, 그 체인을 모두에게 알림
4. 모두는 체인이 맞는지 확인하고 똑같이 매담

블록체인 알고리즘: Proof of Stake

1. 새 데이터가 뜸
2. 대표+검증단이 돈 건 액수에 비례하는 비율에 따라 무작위로 선발됨
3. 대표가 확인하고 현재 블록(거래내역)을 매달고, 모두에게 알림
4. 모두는 블록이 맞는지 확인하고 똑같이 매담.

블록체인 알고리즘 (4/4)

기록된 것은 누구도 고치지 못해야

- A가 (A가, B에게, 500원)이 쓰여졌던 예전 블락을 (A가, B에게, 5원)으로 고치려한다.

블록체인 알고리즘 (4/4)

기록된 것은 누구도 고치지 못해야

- A가 (A가, B에게, 500원)이 쓰여졌던 예전 블락을 (A가, B에게, 5원)으로 고치려한다.
- 방식-I(POW). 해낼 일이 무지무지 많다.
  - 예전 블락 이후의 모든 블락을 다시 써야할텐데.
  - 그 일들에도 대표가되야하는데.

블록체인 알고리즘 (4/4)

기록된 것은 누구도 고치지 못해야

- A가 (A가, B에게, 500원)이 쓰여졌던 예전 블락을 (A가, B에게, 5원)으로 고치려한다.
- 방식-I(POW). 해낼 일이 무지무지 많다.
  - 예전 블락 이후의 모든 블락을 다시 써야할텐데.
  - 그 일들에도 대표가되야하는데.
- 방식-II(POS). 돈비례로 대표+검증단이 무작위로 결정되는데.
  - 대표+검증단이 누가될지 모르므로 매수불가.
  - 대표+검증단이 틀리면 돈 건걸 날릴수있다.

블록체인 알고리즘 요소 기술 (1/3)

암호기술(cryptography): 진품감정에 필요

- (A가, B에게, 500원) 거래가 뜨면, 정말 A 건가?
- 대표가 검증한 블락이 뜨면, 정말 대표 건가?

블록체인 알고리즘 요소 기술 (2/3)

해쉬함수(hashing function): 일거리내기 + 블락매달기 + 변경불가에 필요

해쉬함수: 결과공간이 유한집합인 함수

유용하려면, 가능한한: $\forall x \neq x'.\text{hash}(x) \neq \text{hash}(x')$

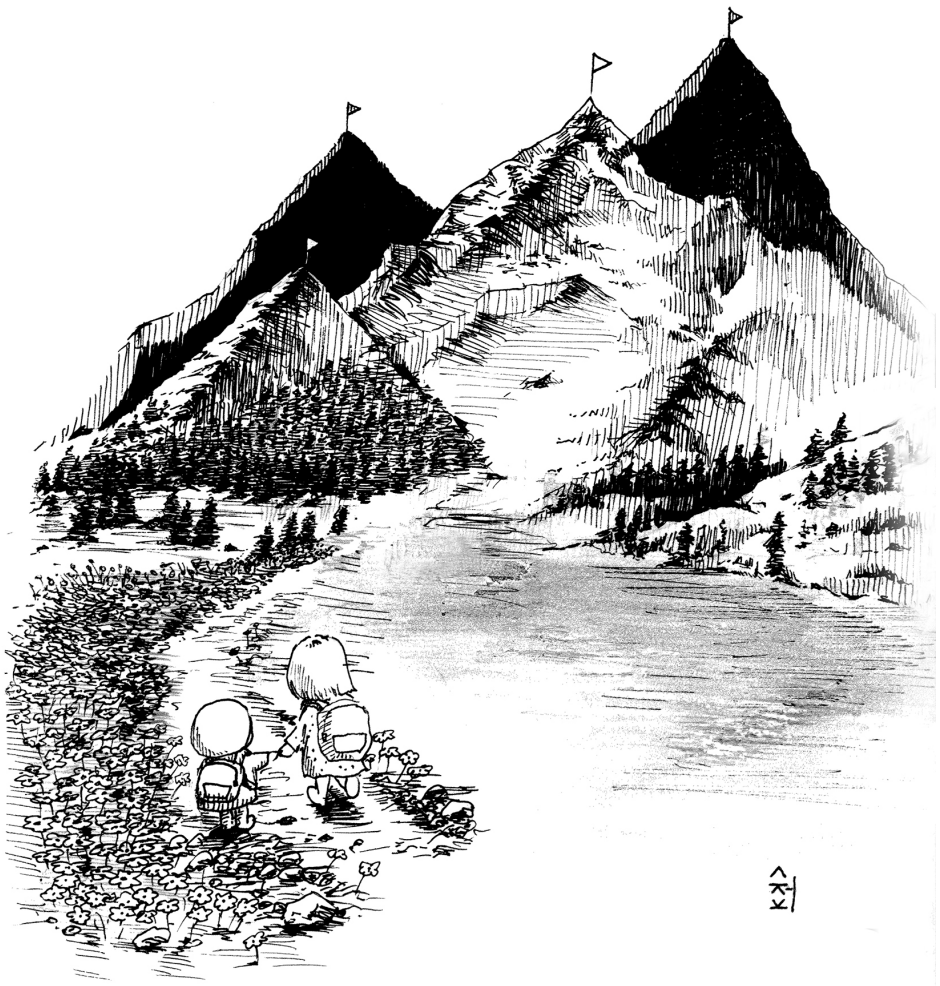
- 방식-I(POW, Proof of Work)에서
- 일거리: 특별한 n 을 찾아라.
  - 예) 해쉬값 $\text{hash}(n + B)$이 000으로 시작하는. (B는 상수 = 이전블락 해시값 + 현재블락 데이터)
- 블락매달기: 체인 $= n + B$
- 변경불가: 블락 변경 $\Rightarrow$ 그 블락 이후의 체인 모두 다시 찾아야

블록체인 알고리즘 요소 기술 (3/3)

무작위함수(pseudo-random number generation function):
블락매달기 + 변경불가에 필요

- 방식-II(POS, Proof of Stake)에서
- 일거리: 없음.
- 블락매달기: 대표가 제안하고 검증단이 검증. 틀리면 건 돈 날림.
- 변경불가: 블락 내용을 바꾸면, 무작위 선발한 대표+검증단이 거부. 무작위 선발되므로 미리 매수불가.

컴퓨터과학이 여는 세계



여기까지 입니다

컴퓨터라는 마음의 도구

그 기원과 구현
그 도구를 다루는 알고리즘과 언어
그 도구의 응용

컴퓨터과학

컴퓨터과학은
머리로 궁리하는 것(intelligence)에 대한 연구(science)이고
그 응용은 인간 지능/본능/현실의 확장이다.

Computer Science is
the science of intelligence
and its application is extensions of human intl/inst/reality

마치

생명과학이 생명현상에 대한 연구이고
그 응용이 인간 수명의 연장이듯이.

물리가 자연에 대한 연구이고
그 응용이 에너지원의 확장이듯이.

의학이 인간 신체의 연구이고
그 응용이 암 정복이듯이.

□