# The History of AI

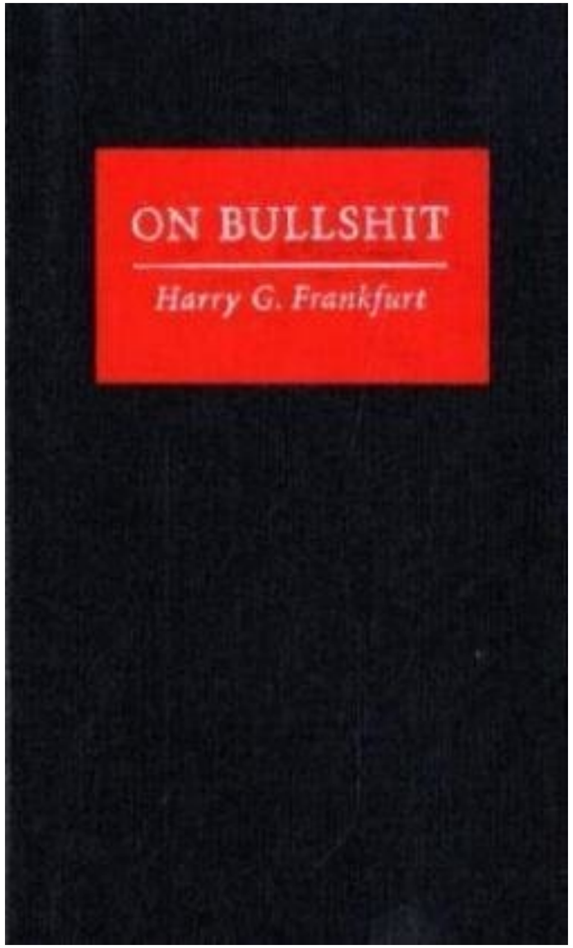
Thomas Haigh, University of Wisconsin—Milwaukee

Senior Digital Humanism Fellow, IWM

# Worshop Description

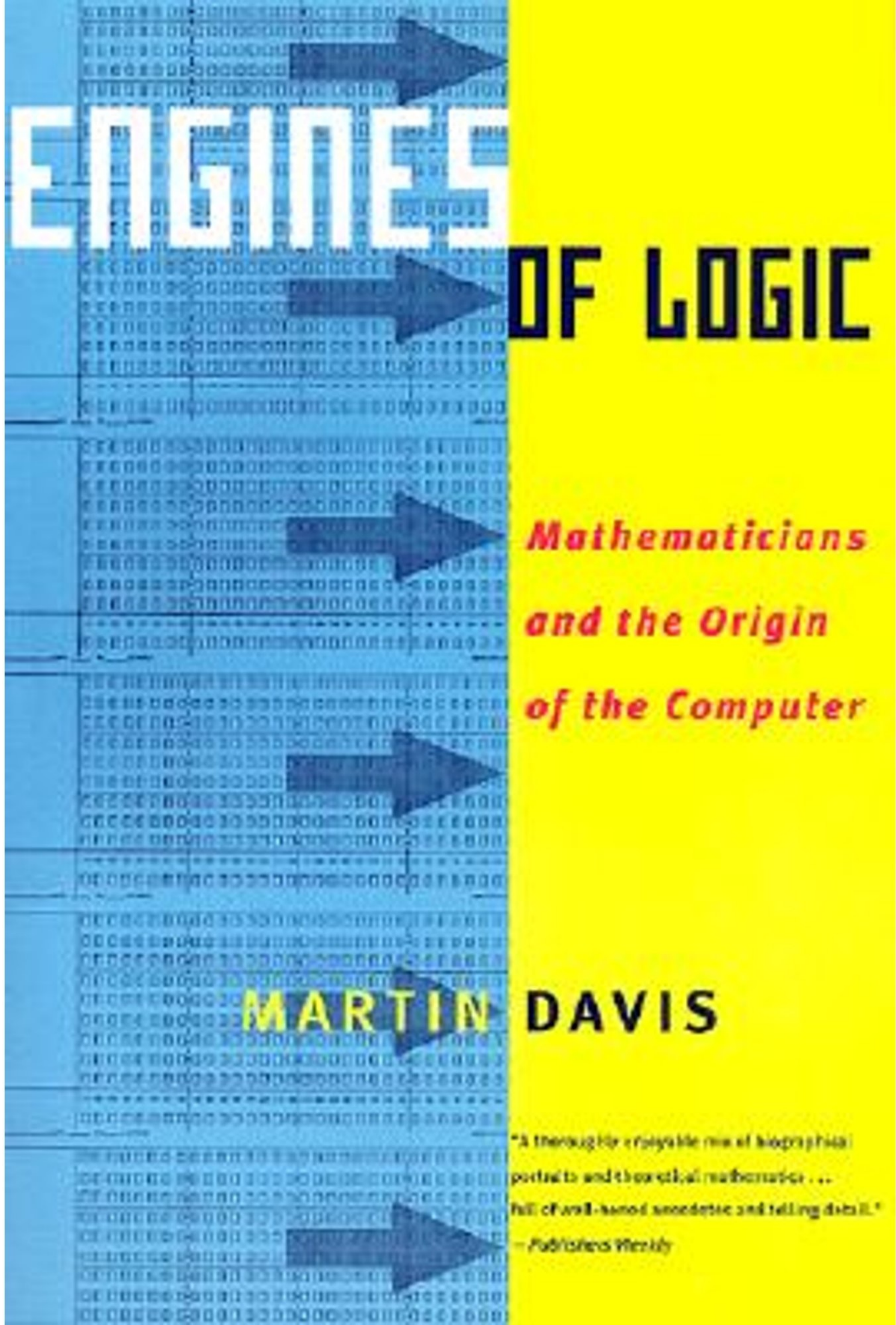
In our workshop we will focus on the challenges for computer science and the sciences in general. We want to discuss and reflect on the possible **paradigm shift** in computer science, **the move from logic and algorithmic certainty to probability and LLMs as story machines**.

# Engines of Logic to Engines of Bullshit?



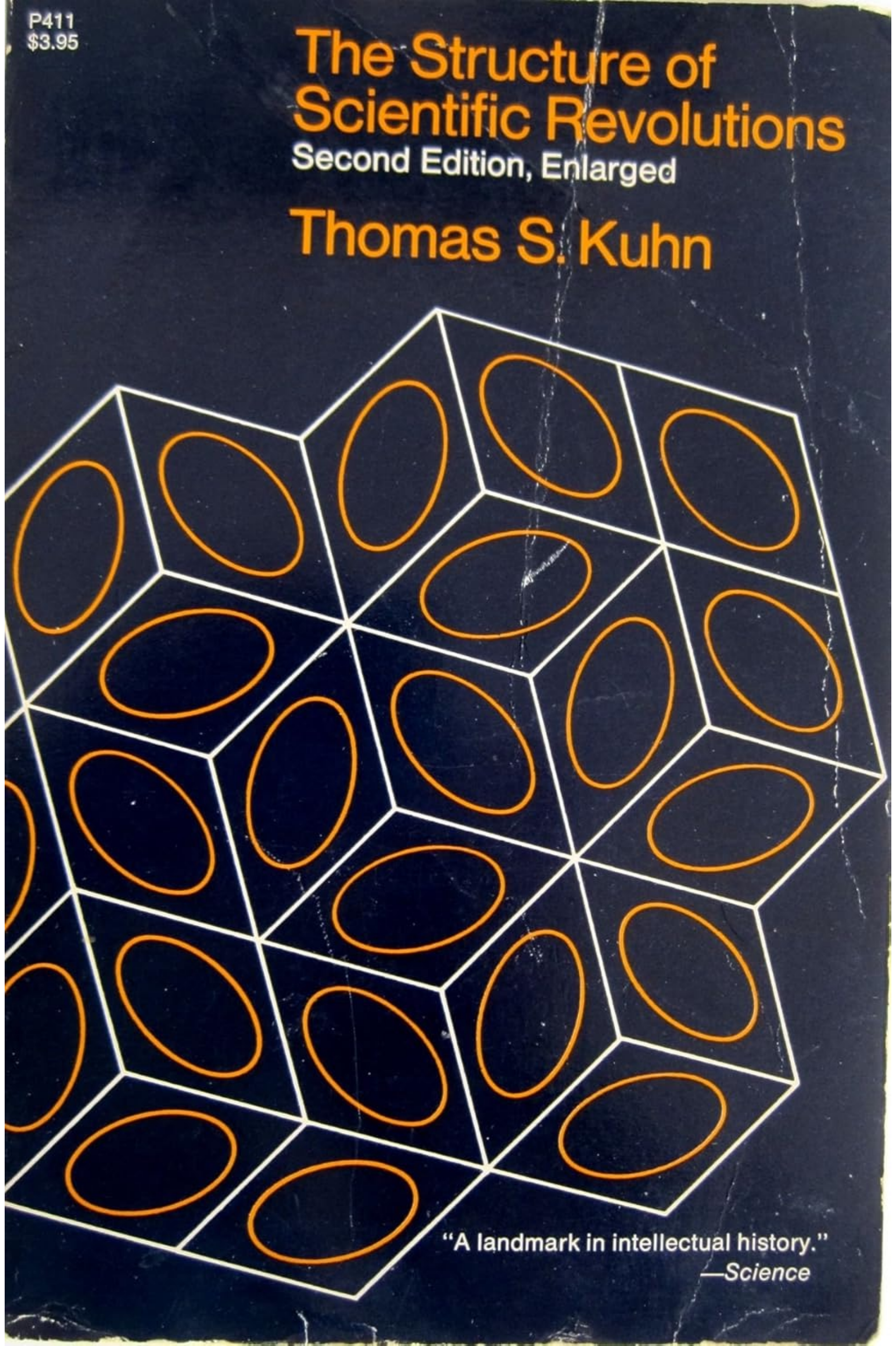

Bullshitting is speaking as with confidence while having no interest in whether the statements being made are true.

Liars have a relationship to the truth, which they are deliberately choosing to disregard, but bullshitters are gloriously untethered by facts.



# New paradigm! What was the old one?

- Paradigm shift, in a "scientific revolution"
  - Which has been unfashionable with historians of science since the 1980s
- Versus "normal science" building incrementally on existing paradigm
- Underlying meaning of paradigm is a tangible example of problem-solving power



# AI history series in *Communications of the ACM*

viewpoints

DOI:10.1145/3593007

Thomas Haigh

## Historical Reflections

## Conjoined Twins: Artificial Intelligence and the Invention of Computer Science

*How artificial intelligence and computer science grew up together.*

HYPE AND HANDWRINGING concerning artificial intelligence (AI) abound. Technologies for face recognition, automatic transcription, machine translation, the generation of text and images, and image tagging have been deployed on an unprecedented scale and work with startling accuracy. Optimists believe the promises of self-driving cars and humanoid robots; pessimists worry about mass unemployment and human obsolescence; critics call for ethical controls on the use of AI and decry its role in the propagation of racism.

Right now, AI refers almost exclusively to neural network systems able to train themselves against large data-sets to successfully recognize or generate patterns. That is a profound break with the approaches behind previous waves of AI hype. In this column, the first in a series, I will be looking back to the origins of AI in the 1950s and 1960s. Artificial intelligence was born out of the promise that computers would quickly outstrip the ability of human minds to reason and the claim that building artificial minds would shed light on human cognition. Although the deep learning techniques underlying today's systems are relatively new, artificial intelligence was a key component in the emergence of computer science as an academic discipline.

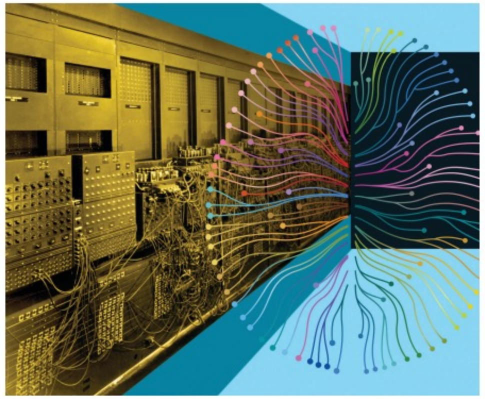


IMAGE BY ANDRIJ BORYS ASSOCIATES, USING PHOTO FROM EVERETT COLLECTION/SHUTTERSTOCK

### Giant Cybernetic Brains

More than commonly realized, the modern computer was itself viewed as a thinking machine within the rich stew of what was about to be branded as *cybernetics*. The basic architecture of modern computers, centered on the retrieval of numerically coded instructions from an addressable high-speed store, was first described in John von Neumann's "First Draft of a Report on the EDVAC." As von Neumann wrote this material in early 1945 he was enmeshed in discussions with a group attempting to charter a "Teleological Society" to explore the radical idea that organisms and machines were substantively equivalent. Von Neumann described the building blocks of digital computer logic, later known as *gates*, with the biological term *neurons*. This was inspired by the work of Warren McCulloch and Walter Pitts, who had asserted that real neurons worked



opinion

DOI:10.1145/3625833

Thomas Haigh

## Historical Reflections

## There Was No 'First AI Winter'

*Despite challenges and failures, the artificial intelligence community grew steadily during the 1970s.*

AS I CONCLUDED my June Historical Reflections column, artificial intelligence had matured from an intellectual brand invented to win funding for a summer research workshop to one of the most prestigious fields in the emerging discipline of computer science. Four of the first 10 ACM A.M. Turing Award recipients were AI specialists: Marvin Minsky, Herb Simon, Allen Newell, and John McCarthy. These men founded the three leading AI labs and played central roles in building what are still the top three U.S. computer science programs at MIT, Stanford, and Carnegie Mellon. Conceptually AI was about uncovering and duplicating the processes behind human cognition; practically it was about figuring out how to program tasks that people could do but computers could not. Although *connectionist* approaches based on training networks of simulated neurons had been prominent in the primordial stew of cybernetics and automata research from which AI emerged, all four Turing Award recipients favored the rival symbolic approach, in which computers algorithmically manipulated symbols according to coded rules of logic.

### A History of Failed Ideas?

AI was born in hype, and its story is usually told as a series of cycles of fervent enthusiasm followed by bitter disappointment. Michael Wooldridge, himself an eminent AI researcher, began

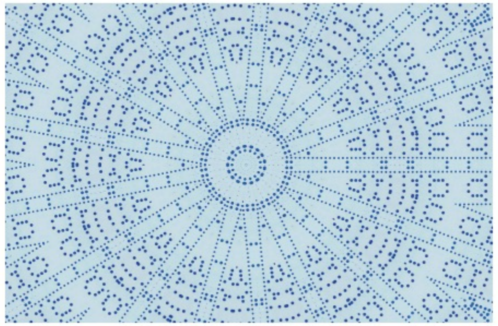


IMAGE BY BRAVISSIMOS

his recent introduction to the field by remembering when he told a colleague about his plan to tell "the story of AI through failed ideas." In response, "she looked back at me, her smile now faded. 'It's going to be a bloody long book then.'"[22]

Major awards lag years behind research. By the time Newell and Simon shared the 1975 ACM A.M. Turing Award the feasibility of their approaches to AI was being increasingly challenged. The AI community would have to wait 19 years for another winner. It was displaced as the intellectual high ground of the emerging discipline by *theoretical computer science*, a field centered on mathematical analysis of algorithms, which garnered nine awardees during the same period.[a] This new focus bolstered the intellectual respectability of computer science with a body of theory that was impeccably mathematical yet unlike numerical analysis, which was falling out of computer science over the same period, not directly useful to or understood by other schol-

a These awards focused on computational complexity theory and the analysis of algorithms. I am construing theoretical computer science here to encompass the work of Rabin and Scott (1976), Cook (1982), Karp (1985), Hopcroft and Tarjan (1986), Milner (1991), and Hartmanis and Stearns (1993). I am not including winners cited primarily for contributions to programming languages, except for Milner whose citation emphasized theory, though Wirth and Hoare both made important theoretical contributions.



opinion

DOI:10.1145/3634901

Thomas Haigh

## Historical Reflections

## How the AI Boom Went Bust

*Fallout from an exploding bubble of hype triggered the real AI Winter in the late 1980s.*

IN MY LAST two columns (June 2023 and December 2023) I followed the history of artificial intelligence (AI) as an intellectual brand and subfield of computer science, from its creation in 1955 through to the end of the 1970s. While acknowledging that AI faced high-profile skepticism from the mid-1960s onward, I argued the 1970s were a time of steady growth for the AI research community. Contrary to popular belief, the "first AI winter" of the 1970s never happened. The 1980s, in contrast, saw the rapid inflation of a government-funded AI bubble centered on the expert system aproach, the popping of which began the real AI winter: a two-decade slump. I will tell that story here, but first I want to say something about how the maturation of AI played out in textbooks and in the computer science curriculum.

### AI in the Curriculum

AI researchers dominated the first 10 years of ACM's A.M Turing Award, suggesting AI initially occupied the intellectual high ground of computer science. Looking at the computer science curriculum hints at a different story, in which AI moved from a marginal subject in the initial degree programs of 1960s to a core field by the end of the 1980s. The history of computer science education remains understudied, but we can get a fuzzy sense of developments by looking at the evolution of ACM's recommended curricula.[2] These recommendations

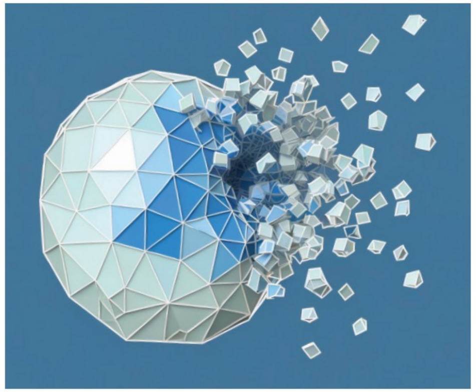


IMAGE CREATED BY ANDRIJ BORYS ASSOCIATES WITH THE ASSISTANCE OF SHUTTERSTOCK.AI

have a complex relationship to actual practice. Likely they were most closely followed by mid-tier institutions, able to hire across a range of specialties but less likely than Stanford or MIT to have the confidence to build their own unique models around in-house expertise. The first ACM model curriculum, from 1968, described 22 undergraduate courses, including one on "artificial intelligence and heuristic programming." As an advanced "methodology" elective this was recommended only for masters' students and for undergraduates pursuing a concentration in theoretical computer science (one of six sample concentrations).[a] The course description suggested a lack of faith in the intellectual maturity of AI: "As this course is essentially descriptive, it might well be taught by surveying various cases of accomplishment in the areas under study."

A decade later, the Curriculum '78 working group recommended an elective covering "basic concepts and techniques," in AI with knowledge representation, search, and system

a https://bit.ly/47b8cmu



June 2023

December 2023

February 2024

opinion

DOI:10.1145/3688379

Thomas Haigh

# Historical Reflections

## Between the Booms: AI in Winter

*After people stopped caring, artificial intelligence got much more interesting.*

OBSERVING THE TSUNAMI of artificial intelligence (AI) hype that has swept over the world in the past few years, science fiction writer Ted Chiang staked out a contrarian position. "Artificial intelligence," he insisted, was just a "poor choice of words ... back in the '50s" that had caused "a lot of confusion." Under the rubric of intelligence, verbs such as "learn," "understand," and "know" had been misappropriated to imply sentience where none existed. The right words, he suggested, would have been "applied statistics." Chiang was correct that AI has always been a fuzzy term used to market specific technologies in a way that has little inherent connection to cognition. It is also true that most current AI-branded technologies work by modeling the statistical properties of large training datasets.

But Chiang's implication that AI has been consistently and uniformly statistical since the 1950s is quite wrong. The approaches that dominated the field from the 1960s to the 1980s owed nothing whatsoever to statistics or probability. In this column, I look at the shift of AI research toward probabilistic methods and at the revival of neural networks. It is a complicated story, because the shift toward probabilistic methods in AI was not initially driven by neural networks, and the revival of neural networks was, until recently, more likely to be branded as machine learning than as AI.

As I explained in my last column, 20th century interest in AI peaked in the 1980s, driven by enthusiasm for expert systems and a flood of public money. AI was moving for the first time beyond the laboratory and into a swarm of startup companies and research groups in large corporations. Then the bubble burst and the famous AI winter set in.

> **The shift was brutal, as changes in technological fashion often are.**

The shift was brutal, as changes in technological fashion often are. Nobody wanted to fund startups anymore, as sales of their products and services slumped. System development groups inside companies could no longer expect that associating themselves with AI would win resources and respect, though some continued under other names. In the 1980s, anything that automated complex processes by applying encoded rules had been called an expert system. The same basic idea was rebranded as business logic during the 1990s as part of the push for distributed computer architectures. Rule-based automation was also central to the emerging field of network security.

Universities shift more slowly. I have seen no evidence that courses in AI disappeared from the curriculum or that established AI faculty decamped



November 2024

Final part, either
"AI Then and Now"
or
"Engines of Bullshit"

Feb 2025?

# How did AI and CS center on “algorithmic certainty” in the first place?

(main part of talk)

# Dartmouth Summer Research Project, 1956

- Proposal funded by the Rockefeller Foundation
- Approx 20 people attended, most for short periods
- Including four men later memorizalied as the cofounders of AI
  - John McCarthy (right rear)
  - Herbert Simon
  - Allen Newell
  - Marvin Minsky (center rear)

A PROPOSAL FOR THE
DARTMOUTH SUMMER RESEARCH PROJECT
ON ARTIFICIAL INTELLIGENCE

J. McCarthy, Dartmouth College
M. L. Minsky, Harvard University
N. Rochester, I.B.M. Corporation
C.E. Shannon, Bell Telephone Laboratories

August 31, 1955

The following are some aspects of the artificial intelligence problem:

1 **Automatic Computers**

If a machine can do a job, then an automatic calculator can be programmed to simulate the machine. The speeds and memory capacities of present computers may be insufficient to simulate many of the higher functions of the human brain, but the major obstacle is not lack of machine capacity, but our inability to write programs taking full advantage of what we have.

2. **How Can a Computer be Programmed to Use a Language**

It may be speculated that a large part of human thought consists of manipulating words according to rules of reasoning and rules of conjecture. From this point of view, forming a generalization consists of admitting a new word and some rules whereby sentences containing it imply and are implied by others. This idea has never been very precisely formulated nor have examples been worked out.

3. **Neuron Nets**

How can a set of (hypothetical) neurons be arranged so as to form concepts. Considerable theoretical and experimental work has been done on this problem by Uttley, Rashevsky and his group, Farley and Clark, Pitts and McCulloch, Minsky, Rochester and Holland, and others. Partial results have been obtained but the problem needs more theoretical work.

4. **Theory of the Size of a Calculation**

If we are given a well-defined problem (one for which it is possible to test mechanically whether or not a proposed answer is a valid answer) one way of solving it is to try all possible answers in order. This method is inefficient, and to exclude it one must have some criterion for efficiency of calculation. Some consideration will show that to get a measure of the efficiency of a calculation it is necessary to have on hand a method of measuring the complexity of calculating devices which in turn can be done if one has a theory of the complexity of functions. Some partial results on this problem have been obtained by Shannon, and also by McCarthy.

5. **Self-Improvement**

Probably a truly intelligent machine will carry out activities which may best be described as self-improvement. Some schemes for doing this have been proposed and are worth further study. It seems likely that this question can be studied abstractly as well.

6. **Abstractions**

A number of types of "abstraction" can be distinctly defined and several others less distinctly. A direct attempt to classify these and to describe machine methods of forming abstractions from sensory and other data would seem worthwhile.

7. **Randomness and Creativity**

A fairly attractive and yet clearly incomplete conjecture is that the difference between creative thinking and unimaginative competent thinking lies in the injection of a some randomness. The randomness must be guided by intuition to be efficient. In other words, the educated guess or the hunch include controlled randomness in otherwise orderly thinking.

# Institutionalizing AI

- Three earliest centers for AI research in the US
  - MIT (Minsky, McCarthy) in 1958“with two programmers, a secretary, a typewriting machine and six graduate students.”
  - Stanford (McCarthy) AI Project 1962, SAIL in 1965, CS Dept in 1965 & Stanford Research International
  - Carnegie Mellon (Newell & Simon), CS Dept 1965
- Computer Science programs & depts develop in parallel with AI
  - DOD ARPA is biggest funder of computing projects in 1960s
  - AI labs receive extensive funding, esp. MIT with Project MAC
  - First CS PhD graduates in 1968
  - AI culture centers on system building over theory for doctoral projects
- According to most rankings, the top global CS departments as of 2024 are
  1. MIT
  2. Stanford
  3. Carnegie Mellon

# The AI agenda of the 1960s to 1980s

- Intelligence = things computers cannot *currently* do
- Symbolic approach wins out over neural networks
  - Computers manipulate symbols representing knowledge of the world according to algorithmic rules
  - Search as the core technique
- Effort to find general purpose reasoning methods that work across tasks
  - Though specific tasks were selected as test cases
- Assumption that the human brain works the same way
- By the 1970s, increasing focus on knowledge representation and its difficulties
  - In part as a response to criticisms of naive focus on reasoning methods by Joseph Weizenbaum, Herbert Dreyfus, and others

# Symbolic AI Concepts

- Search
- Heuristics
- Rewriting of Symbols
- Structured knowledge representation
- Emphasis on implementation over theory

# Origins of Computer Science

- 1950s: many leading universities have
  - Computer building efforts in electrical engineering groups
  - Campus computer centers, to support scientific work
- Circa 1960, first academic programs and then departments in computer science
  - Computing as area of study in its own right
  - Which means it needs its own areas of theory
- Faculty and expertise come from existing disciplines. Including
  - Electronic engineering
  - Numerical analysis
  - Discrete mathematics
  - Information theory
  - Programming languages & compilers

# Tears of Donald Knuth

- One slide – point is CS history not being written by either side.

**Historical work on computing as a contribution to computer science versus as a contribution to a subfield of history or social science studies.**

| | | |
|---|---|---|
| **Historical work on computing framed primarily as a contribution to computer science.** | Common in the 1970s and 1980s, less so today. | Almost impossible to accomplish (Campbell-Kelly's early work being an exception). |
| **Historical work on computing framed primarily as a contribution to a subfield of history or science studies.** | Fairly common, particularly for trained historians of computing working outside academia or in fields where history is not seen as central. | Increasingly common, with hopeful signs for further growth. |
| | **Historical work on computing as a secondary interest or activity during retirement.** | **Historical work on computing as the major focus of a scholarly career, for which one is hired or promoted.** |

V viewpoints

DOI:10.1145/2688497

Thomas Haigh

## Historical Reflections
# The Tears of Donald Knuth

*Has the history of computing taken a tragic turn?*

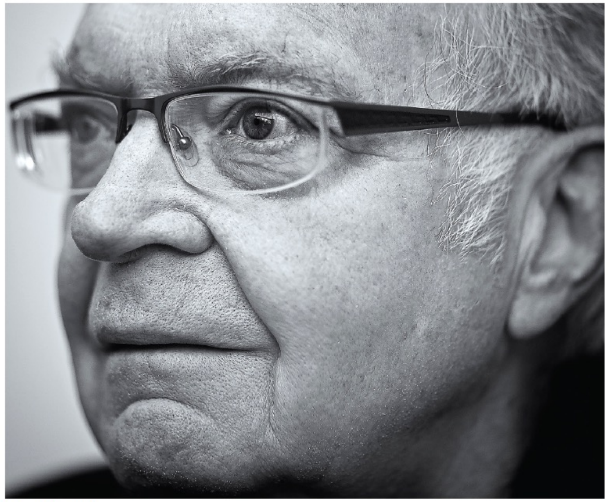
IN THIS COLUMN I will be looking at the changing relationship between the discipline of computer science and the growing body of scholarly work on the history of computing, beginning with a recent plea made by renowned computer scientist Donald Knuth. This provides an opportunity to point you toward some interesting recent work on the history of computer science and to think more broadly about what the history of computing is, who is writing it, and for whom they are writing.

Last year historians of computing heard an odd rumor: that Knuth had given the Kailath lecture at Stanford University and spent the whole time talking about us. Its title, "Let's Not Dumb Down the History of Computer Science," was certainly intriguing, and its abstract confirmed that some forceful positions were being taken.[a] The online video eventually showed something remarkable: his lecture focused on a single paper, Martin Campbell-Kelly's 2007 "The History of the History of Software."[6,b] Reading it had deeply saddened Knuth, who "finished reading it only with great difficulty" through his tear-stained glasses.

### What Knuth Said

Knuth began by announcing that, despite an aversion to confrontation, he would be "flaming" historians of computing. This, he worried "could turn out to be the biggest mistake of my life." The bout might nevertheless be seen as a mismatch. Knuth is among the world's most celebrated computer scientists, renowned for his ongoing project to classify and document families of algorithms in *The Art of Computer Programming* and for his creation of the TeX computerized typesetting system ubiquitous within computer science and mathematics. Campbell-Kelly has a similar prominence within the much smaller community of historians of computing but, even by Google Scholar's generous definitions, the paper that saddened Knuth has been cited only nine times.

Knuth then enumerated his motivations, as a computer scientist, to read the history of science. First, reading history helped him to understand the process of discovery. Second, understanding the difficulty and false starts experienced by brilliant historical scientists in making discoveries that specialists now find obvious helped him to



Donald Knuth.

PHOTOGRAPH BY RAJAN P. PARRIKAR

a See http://kailathlecture.stanford.edu/featured_speaker.html#abstract_bio.
b The video is posted at http://kailathlecture.stanford.edu/2014KailathLecture.html.

# AI was intertwined with Computer Science

- Minsky, McCarthy, Newell, Simon are among first 11 ACM Turing Award winners
- Which suggests recognition of AI as the most prestigious area of the new discipline
- I conceive of computer science as an unusually federated discipline…

(1975)
Newell, Allen *
Simon, Herbert ("Herb") Alexander *

(1974)
Knuth, Donald ("Don") Ervin

(1973)
Bachman, Charles William *

(1972)
Dijkstra, Edsger Wybe *

(1971)
McCarthy, John *

(1970)
Wilkinson, James Hardy ("Jim") *

(1969)
Minsky, Marvin *

(1968)
Hamming, Richard W*

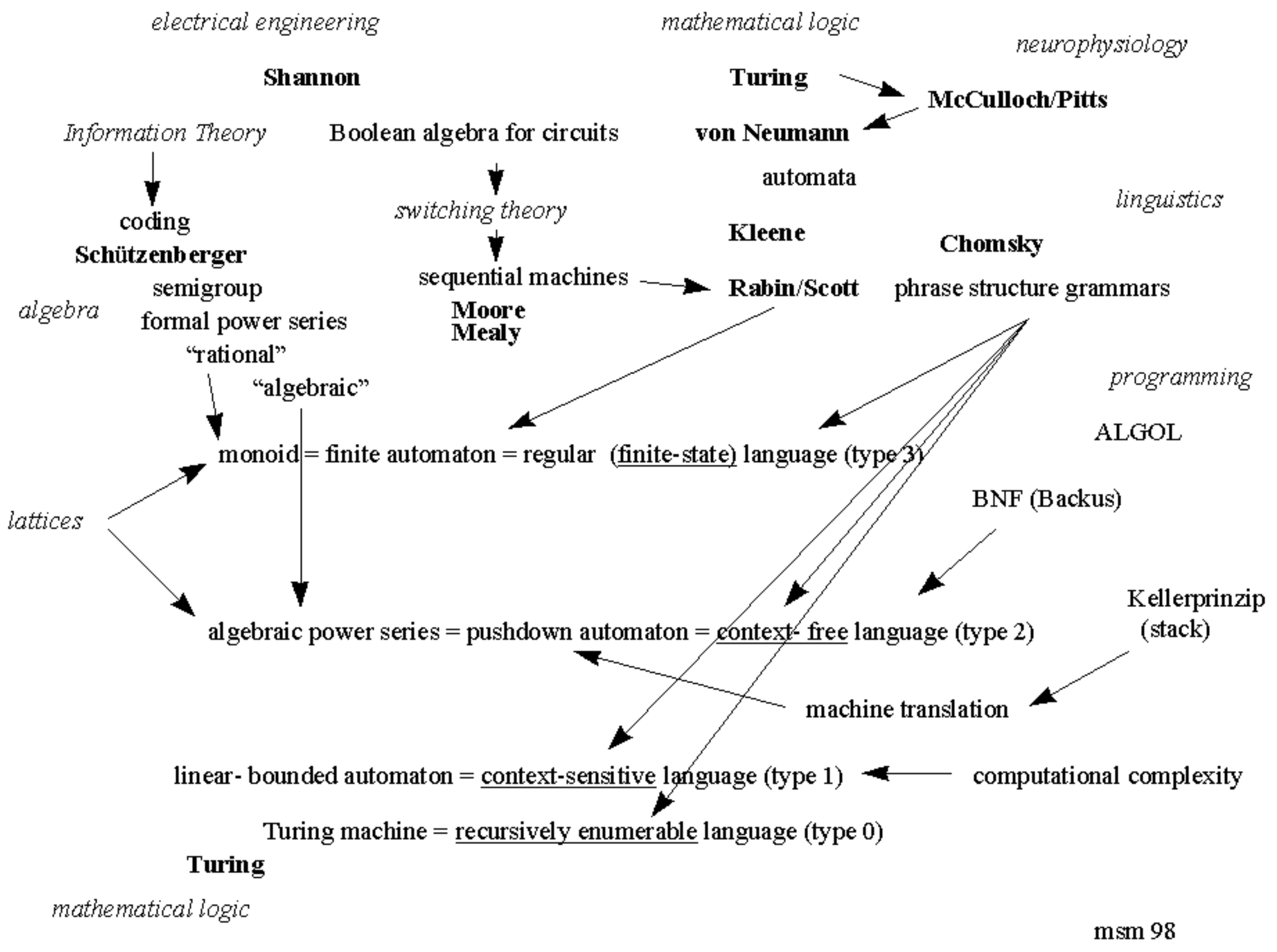
(1967)
Wilkes, Maurice V.*

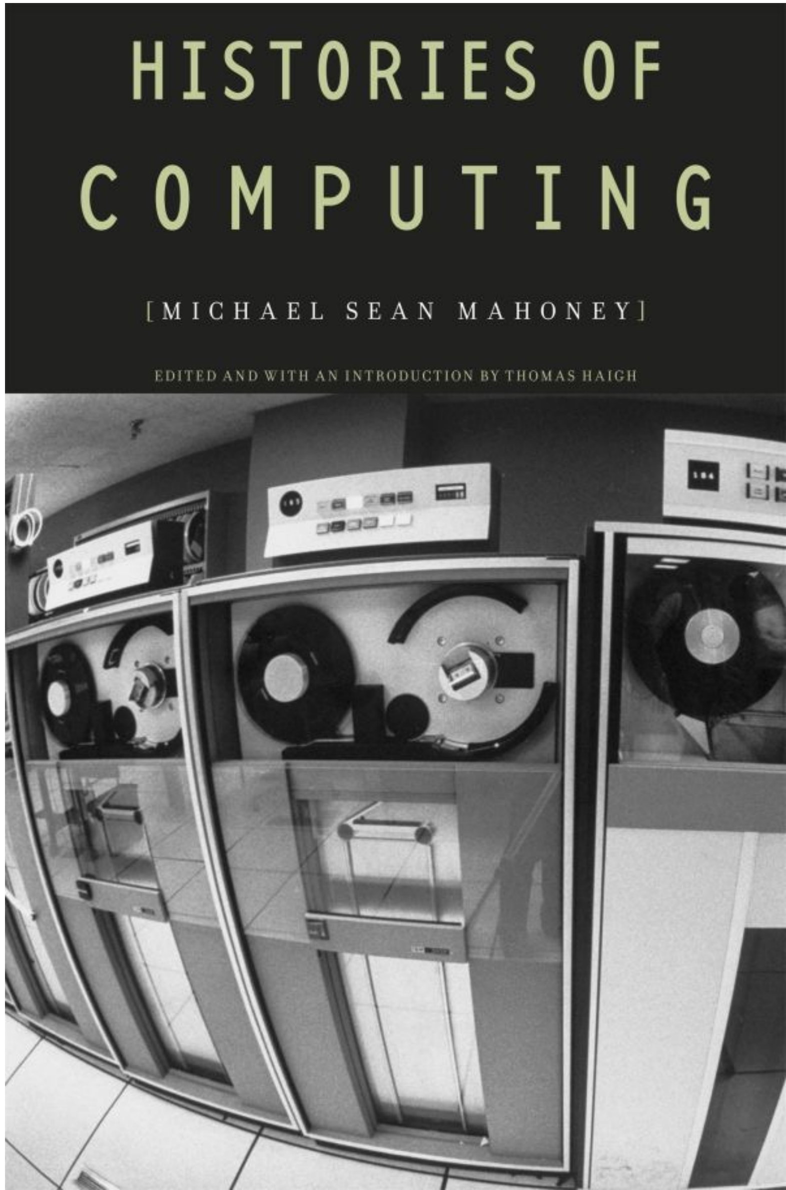
(1966)
Perlis, Alan J *

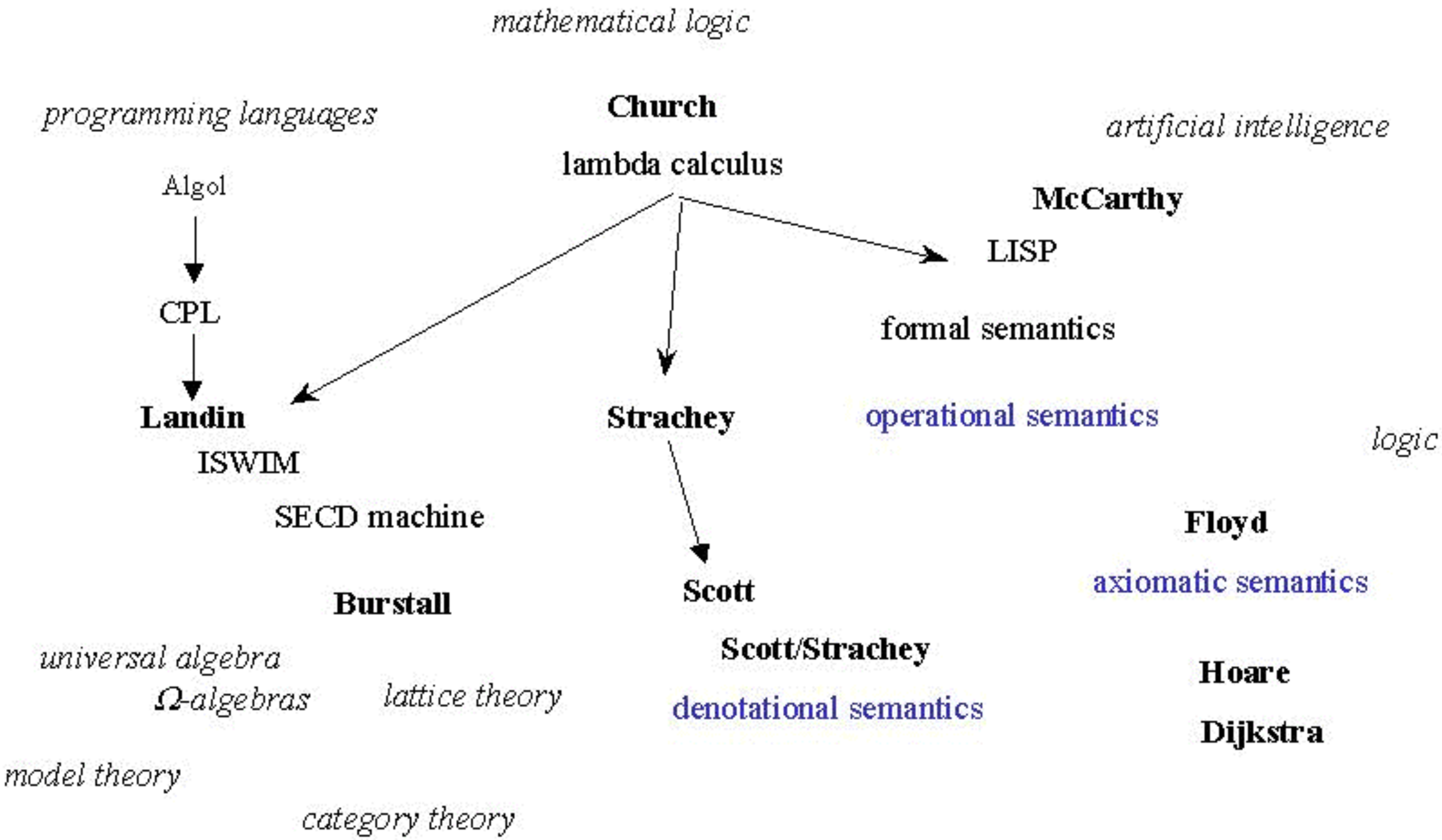
# Theoretical Computer Science

- No AI winners between 1975 and 1994
- Awards made to
  - Rabin & Scott (1976) for their joint paper "Finite Automata and Their Decision Problem"
  - Cook (1982) for "The Complexity of Theorem Proving Procedures" and NP completeness
  - Karp (1985) for most notably, contributions to the theory of NP-completeness.
  - Hopcroft & Tarjan (1986), for fundamental achievements in the design and analysis of algorithms and data structures.
  - Milner (1991), (three distinct things mentioned in citation, incl the study of the relationship between operational and denotational semantics.
  - Hartmanis & Stearns (1993), established the foundations for the field of computational complexity theory.
- Scoreline: Theory 9, AI 4.

# Assembling Computer Science (Mahoney)




mathematical logic
Church
lambda calculus
programming languages
Algol
CPL
Landin
ISWIM
SECD machine
artificial intelligence
McCarthy
LISP
formal semantics
Strachey
operational semantics
logic
Floyd
axiomatic semantics
Scott
Burstall
Scott/Strachey
universal algebra
Ω-algebras
lattice theory
denotational semantics
Hoare
Dijkstra
model theory
category theory

# AI in Early CS Curriculum

- Computer science as a federated discipline
  - ACM SIGs appear in late-1960s
  - First model curriculum 1968
- AI is a obscure elective in 1968 (M.Sc. students & theory specialists)
- 1978: LISP in core course, standard undergraduate AI elective
- 1988: AI  & Robotics as one of nine core areas of computer science

CURRICULUM 68

Recommendations for Academic Programs in Computer Science

A Report of the ACM Curriculum Committee on Computer Science

*Dedicated to the Memory of Silvio O. Navarro*

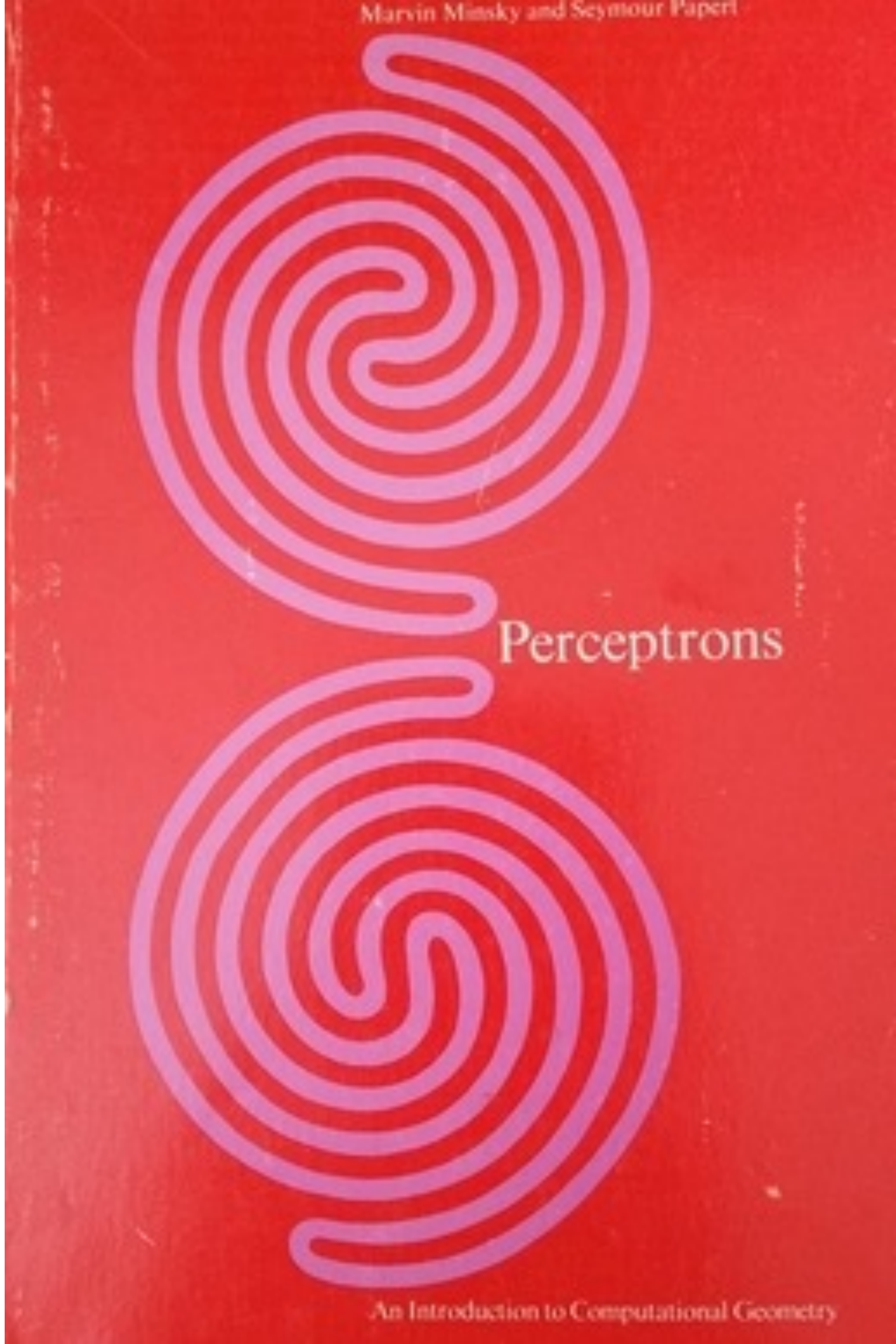
This report contains recommendations on academic programs in computer science which were developed by the ACM Curriculum Committee on Computer Science. A classification of the subject areas contained in computer science is presented and twenty-two courses in these areas are described. Prerequisites, catalog descriptions, detailed outlines, and annotated bibliographies for these courses are included. Specific recommendations which have evolved from the Committee's 1965 Preliminary Recommendations are given for undergraduate programs. Graduate programs in computer science are discussed, and some recommendations are presented for the development of master's degree programs. Ways of developing guidelines for doctoral programs are discussed, but no specific recommendations are made. The importance of service courses, minors, and continuing education in computer science is emphasized. Attention is given to the organization, staff requirements, computer resources, and other facilities needed to implement computer science educational programs.

KEY WORDS AND PHRASES: computer science courses, computer science curriculum, computer science education, computer science academic programs, computer science graduate programs, computer science undergraduate programs, computer science course bibliographies

*CR* CATEGORIES: 1.52

# Minsky & Papert, Perceptrons

- Published 1969
- Minsky's dramatic disavowal of neural nets
  - Focused on one simple variety
  - But also critical of multi-level networks
- Treated in AI folklore as an important cause of the exclusion of neural net research from AI



# Newell – Series of Binary Choices

- "1955: Symbolic vs. continuous systems, splits AI from cybernetics"
- "1955-65: Problem solving vs. recognition, splits AI from pattern recognition"
  - Though it later "rejoins AI via robotics"

From "Intellectual Issues in the History of Artificial Intelligence" (1983)

Table 1. The Intellectual Issues of AI

| | |
|---|---|
| 1640–1945 | Mechanism versus teleology: settled with cybernetics |
| 1800–1920 | Natural biology versus vitalism: establishes the body as a machine |
| 1870– | Reason versus emotion and feeling #1: separates machines from men |
| 1870–1910 | Philosophy versus the science of mind: separates psychology from philosophy |
| 1910–1945 | Logic versus psychologic: separates logic from psychology |
| 1940–1970 | Analog versus digital: creates computer science |
| 1955–1965 | Symbols versus numbers: isolates AI within computer science |
| 1955– | Symbolic versus continuous systems: splits AI from cybernetics |
| 1955–1965 | Problem-solving versus recognition #1: splits AI from pattern recognition |
| 1955–1965 | Psychology versus neurophysiology #1: splits AI from cybernetics |
| 1955–1965 | Performance versus learning #1: splits AI from pattern recognition |
| 1955–1965 | Serial versus parallel #1: coordinate with above four issues |
| 1955–1965 | Heuristics versus algorithms: isolates AI within computer science |
| 1955–1985 | Interpretation versus compilation: isolates AI within computer science |
| 1955– | Simulation versus engineering analysis: divides AI |
| 1960– | Replacing versus helping humans: isolates AI |
| 1960– | Epistemology versus heuristics: divides AI (minor); connects with philosophy |
| 1965–1980 | Search versus knowledge: apparent paradigm shift within AI |
| 1965–1975 | Power versus generality: shift of tasks of interest |
| 1965– | Competence versus performance: splits linguistics from AI and psychology |
| 1965–1975 | Memory versus processing: splits cognitive psychology from AI |
| 1965–1975 | Problem-solving versus recognition #2: recognition rejoins AI via robotics |
| 1965–1975 | Syntax versus semantics: splits linguistics from AI |
| 1965– | Theorem-proving versus problem-solving: divides AI |
| 1965– | Engineering versus science: divides computer science, including AI |
| 1970–1980 | Language versus tasks: natural language becomes central |
| 1970–1980 | Procedural versus declarative representation #1: shift from theorem-proving |
| 1970–1980 | Frames versus atoms: shift to holistic representations |
| 1970– | Reason versus emotion and feeling #2: splits AI from philosophy of mind |
| 1975– | Toy versus real tasks: shift to applications |
| 1975– | Serial versus parallel #2: distributed AI (Hearsay-like systems) |
| 1975– | Performance versus learning #2: resurgence (production systems) |
| 1975– | Psychology versus neuroscience #2: new link to neuroscience |
| 1980– | Serial versus parallel #3: new attempt at neural systems |
| 1980– | Problem-solving versus recognition #3: return of robotics |
| 1980– | Procedural versus declarative representation #2: PROLOG |

# Physical Symbol System Hypothesis

- Expressed in Newell & Simon Turing Award lecture, titled "Computer science as empirical inquiry: symbols and search"
- Most quoted line: "A physical symbol system has the necessary and sufficient means for general intelligent action." These include
  - A computer programmed in LISP
  - A universal Turing machine
  - A thinking human
- Concept is "join of computability, physical realizability (and by multiple technologies), universality, the symbolic representational of processes (i.e. interpretability), and, finally, symbolic structure and designation."

# Aside: Does “Subsymbolic” Make Sense Anymore?

- To Newell & Simon, all cognition is symbolic
  - Real or simulated neurons are one of many possible physical media for the “physical symbol system”
  - High level symbolic operations have underlying non-symbolic representations
  - But these are interchangeable, the same mental processes can take place in any physical symbol system
- If we do not believe this claim then
  - In what sense are connectionist approaches SUBsymbolic?

# Early Textbooks Entirely Symbolic

- Seven early textbooks 1971-77
  - All eight authors had Stanford or MIT degrees
- First really successful textbook is *Artificial Intelligence* by Patrick Winston (MIT)
  - No mention of neural nets or connectionism
  - Even though Winston specialized in computer vision and machine learning
- Other leading textbook of the era gave two sentences
  - "Although there have been many attempts to build learning programs starting with a random network, none of them have met with any degree of success. For this reason, we will not discuss this approach any further here." Elaine Rich, *Artificial Intelligence* (New York: McGraw-Hill, 1983).

# Neural nets did not vanish

- But they were banished from AI and CS more broadly
- Rebranded as “pattern recognition” and “machine learning”
- Tied to robotics
- (apparently) studied mostly in engineering disciplines

# Preparing for this workshop

- Has made me think more about the conjunction of
  - AI developing as a subfield of computer science
  - AI rejecting connectionism to be symbolic
- Not coincidental
  - Fundamental intellectual connections between symbolic AI and other emerging high-status areas of computer science
  - Even though Newell stressed the peculiarities of AI
    - 1955-1965: Symbols versus numbers, isolates AI within computer science
    - 1955-1965: Heuristics versus algorithms, isolates AI within computer science
    - 1955-85: Interpretation versus compilation, isolates AI within computer science

# Fragments of my broader project

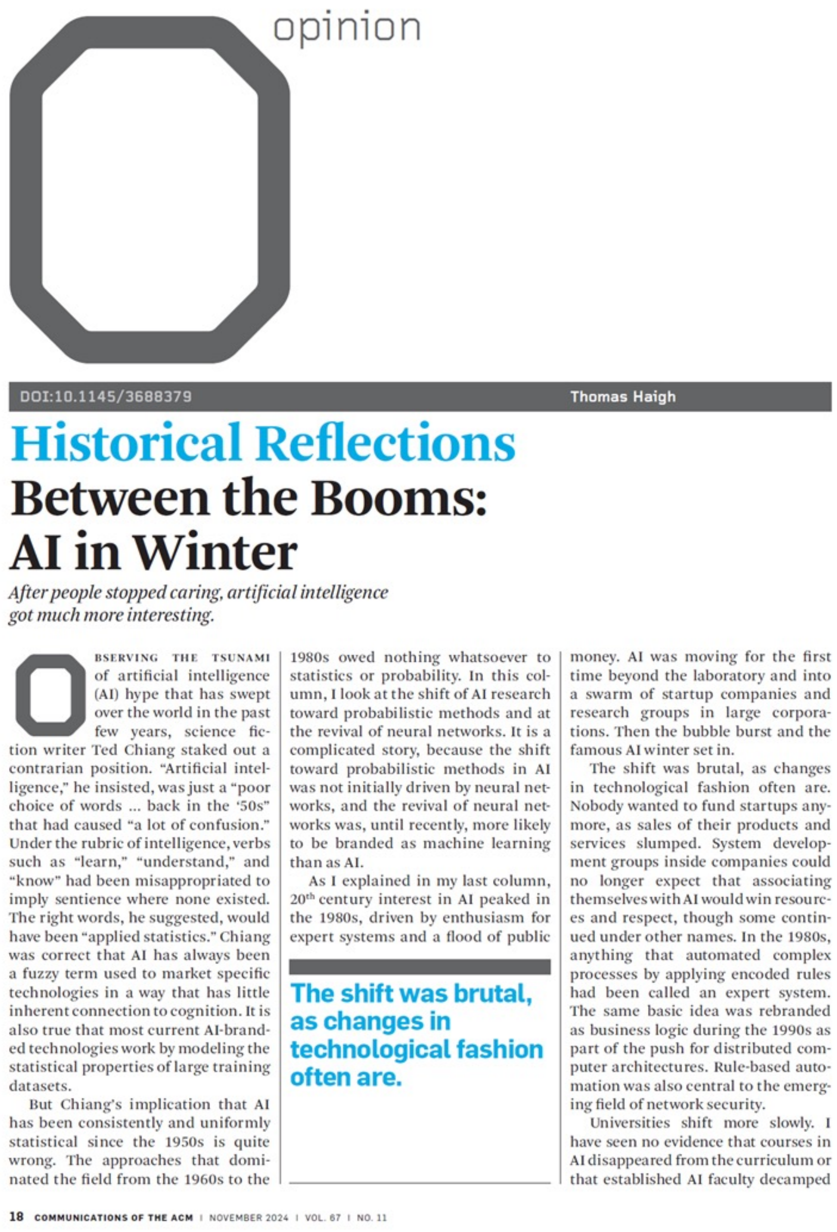
(short concluding epilogue)

# Artificial Intelligence: The History of a Brand

| Chapter | Era | Case Study |
|---|---|---|
| Introduction: The Brand That Wouldn't Die | | (examples of AI hype) |
| 1: The Birth of a Brand | 1950s | GPS (General Problem Solver) |
| 2: Institutionalizing the AI Brand | 1960s | SHRDLU |
| 3: Challenges to the AI Brand | Late 1960s & 1970s | Hearsay II |
| 4: Branding with Knowledge | 1970s | Mycin |
| 5: Selling AI | 1980s | Cyc |
| 6: Out of Fashion, AI Tries New Things | 1990s | Dragon Naturally Speaking |
| 7: Machine Learning Becomes the Hot New Brand | 1980s-2010s | AlexNet |
| 8:Thanks to Chatbots, AI Finally Conquers the World | Present day | ChatGPT |
| Epilogue: What Was AI Anyway? | | |

# Interesting AI Shifts in 1990s

- Complicating the “move from logic and algorithmic certainty to probability and LLMs as story machines”
- First: “Modern AI” (Russell & Norvig) is defined around Bayesian, probabilistic approach pioneered by Judea Pearl.
  - But this is not a big data story
- Second: big data approach comes first in statistical approaches to NLP
  - But this is hidden Markov models, not neural nets
- Third: neural nets are associated mostly with classification, not generation until very recently



opinion

DOI:10.1145/3688379 Thomas Haigh

**Historical Reflections**

**Between the Booms: AI in Winter**

*After people stopped caring, artificial intelligence got much more interesting.*

OBSERVING THE TSUNAMI of artificial intelligence (AI) hype that has swept over the world in the past few years, science fiction writer Ted Chiang staked out a contrarian position. “Artificial intelligence,” he insisted, was just a “poor choice of words … back in the ‘50s” that had caused “a lot of confusion.” Under the rubric of intelligence, verbs such as “learn,” “understand,” and “know” had been misappropriated to imply sentience where none existed. The right words, he suggested, would have been “applied statistics.” Chiang was correct that AI has always been a fuzzy term used to market specific technologies in a way that has little inherent connection to cognition. It is also true that most current AI-branded technologies work by modeling the statistical properties of large training datasets.

But Chiang’s implication that AI has been consistently and uniformly statistical since the 1950s is quite wrong. The approaches that dominated the field from the 1960s to the 1980s owed nothing whatsoever to statistics or probability. In this column, I look at the shift of AI research toward probabilistic methods and at the revival of neural networks. It is a complicated story, because the shift toward probabilistic methods in AI was not initially driven by neural networks, and the revival of neural networks was, until recently, more likely to be branded as machine learning than as AI.

As I explained in my last column, 20th century interest in AI peaked in the 1980s, driven by enthusiasm for expert systems and a flood of public money. AI was moving for the first time beyond the laboratory and into a swarm of startup companies and research groups in large corporations. Then the bubble burst and the famous AI winter set in.

**The shift was brutal, as changes in technological fashion often are.**

The shift was brutal, as changes in technological fashion often are. Nobody wanted to fund startups anymore, as sales of their products and services slumped. System development groups inside companies could no longer expect that associating themselves with AI would win resources and respect, though some continued under other names. In the 1980s, anything that automated complex processes by applying encoded rules had been called an expert system. The same basic idea was rebranded as business logic during the 1990s as part of the push for distributed computer architectures. Rule-based automation was also central to the emerging field of network security.

Universities shift more slowly. I have seen no evidence that courses in AI disappeared from the curriculum or that established AI faculty decamped

18 COMMUNICATIONS OF THE ACM | NOVEMBER 2024 | VOL. 67 | NO. 11

# AI as a Brand

- Reflects continuities over time in
  - Promises made for potential of AI
  - Timeline to realization of superintelligence
  - Threats of AI
- Versus discontinuities in
  - Technologies branded as AI
  - Whether specific approaches are branded as AI or not (e.g. neural nets, speech recognition)
- And even at a specific moment
  - Whether specific technology is branded as AI has more to do with who funds it and where it is developed
  - Less to do with specific technical content
  - Almost nothing to do with any inherent connection to human cognition

# Continuities and Discontinuities

| 20TH CENTURY AI | CONTEMPORARY AI |
|---|---|
| Hugely hyped | Spectacularly hyped |
| Needs fastest computers | Needs fastest computers |
| Applied to arbitrary collection of technologies | Applied to arbitrary collection of technologies |
| Loose connection of tech to cognition | Loose connection of tech to cognition |
| | |
| Mostly academic | Mostly commercial |
| Government funded | Investor funded |
| Symbolic | Connectionist |
| Heuristic search | Statistical prediction |
| Humans usually formulate rules | System trains itself from mass of data |
| Knowledge coded explicitly | Knowledge dispersed over connection weights |
| Rarely applied outside lab | Widely applied on big tech platforms |

# Google Ngram (based on large text corpus)