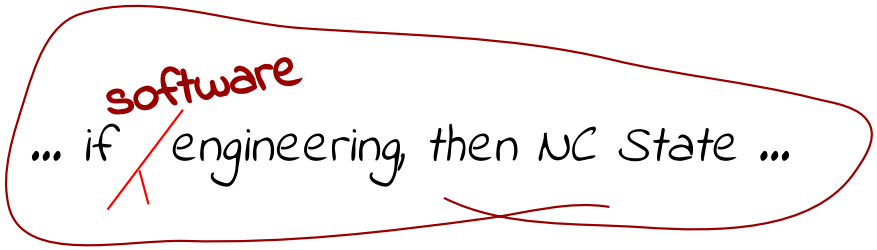


In the age of Software 2.0, what role for software engineers?

Tim Menzies
cs, se, ai, ncstate, usa
timm@ieee.org,

Mar 7, 2019

Association for Computing Machinery

Software 2.0: when software writes itself What does that mean, exactly?

Erik Meijer

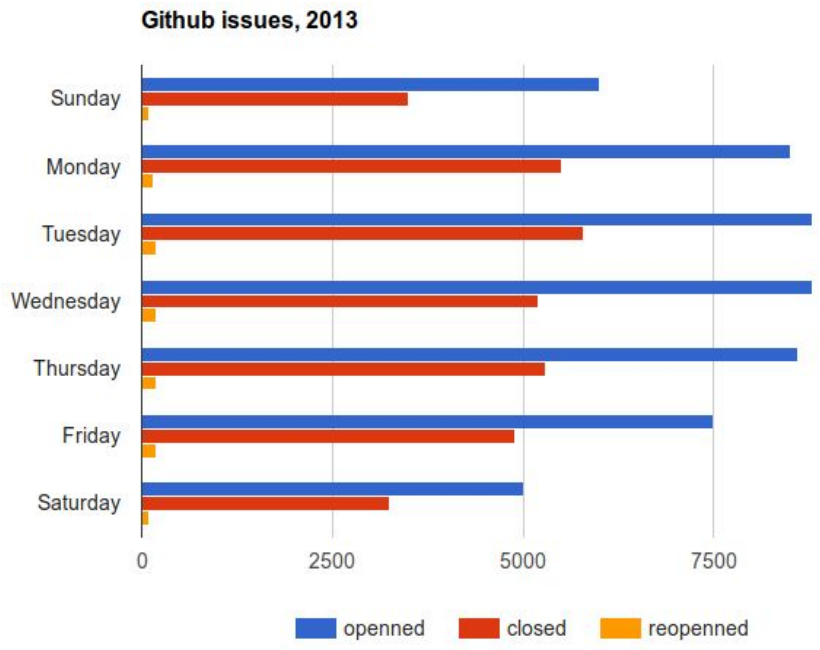
- Behind Every Great Deep Learning Framework Is An Even Greater Programming Languages

- Software 1.0:
  - coffee + human = code

- Software 2.0
  - humans say “what”
  - a agents automatically work out “how”

A counter-view: Programmers > programming

- Still much space for novel human creativity
- Programmers do not just sit in a backroom, just filling in the details to a spec
- Programming is about software engineering
  - The search for feedback on ideas.
  - Half the issues in Github: never closed
  - Programmers spend much time negotiating which, what how, when, who
- Feedback can be handled many ways
  - See below
  - Need humans + automation
    - Not one replacing the other



There are somethings humans just do better than AI

Human factors:

- Convincing you that the system is trustable
- Helping you when the systems fails

Current roles of SE experts
when adopting data mining approaches:

- Problem definition.
- Data collection.
- Data "surrogates" (when the joins fail you)
- Model building.
- **Managing organizational impact**

MINKU, L.L.; MENDES, E.; TURHAN, B. . "Data Mining for SE and Humans in the Loop", *Progress in Artificial Intelligence (PRAI)*, 5(4), 307-314, Nov 2016

Potential roles of SE experts
when adopting data mining approaches :

- Model building involvement.
- Expert domain knowledge.

AI = Not automatic.
A large buffet of options.
Needs skilled human engineers

Can't just throw all problems at huge CPU farms

- Zhe's summer at Google: 3 years of CPU/day
- Wang et al., FSE'13. 15 years of CPU to evaluate software clone configurations
- Truede and Wagner: 30 years of CPU to tune clusters for each corpus

Cannot get humans out of the loop (though some want to try)

- "The notion of user cannot be precisely defined and therefore has no place in CS or SE"
  - Edsger Dijkstra, ICSE 4, 1979

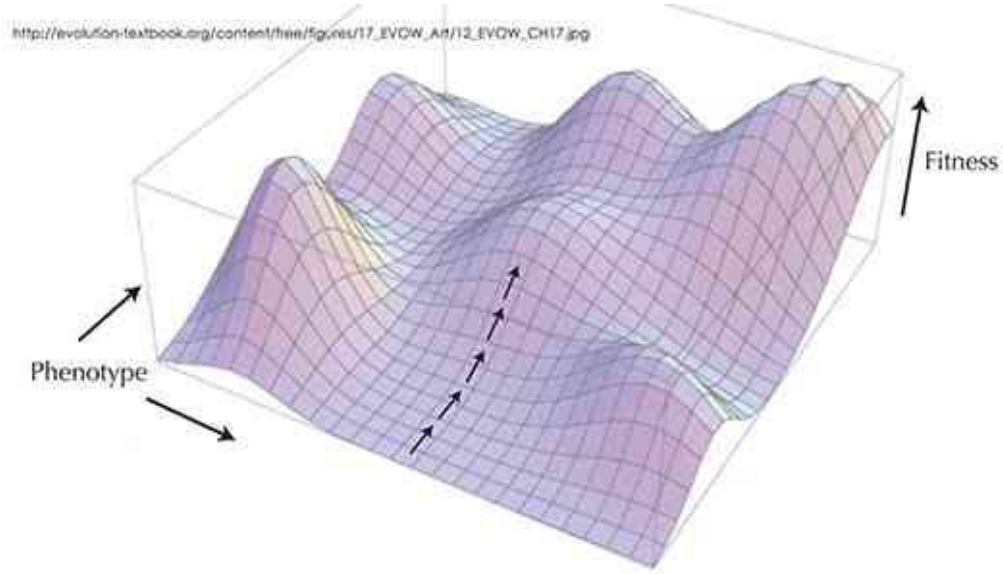
- ".. kill living human experts and resurrect the dead ones"
  - Anonymous machine learning researcher,
  - 1986

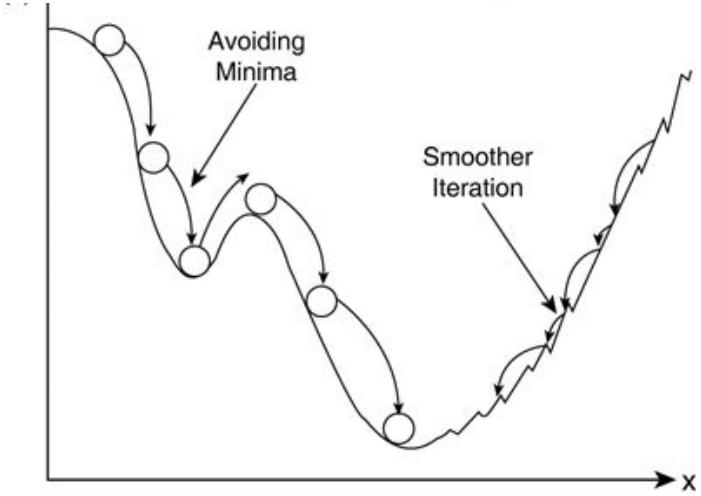
Humans exist. We must work with them

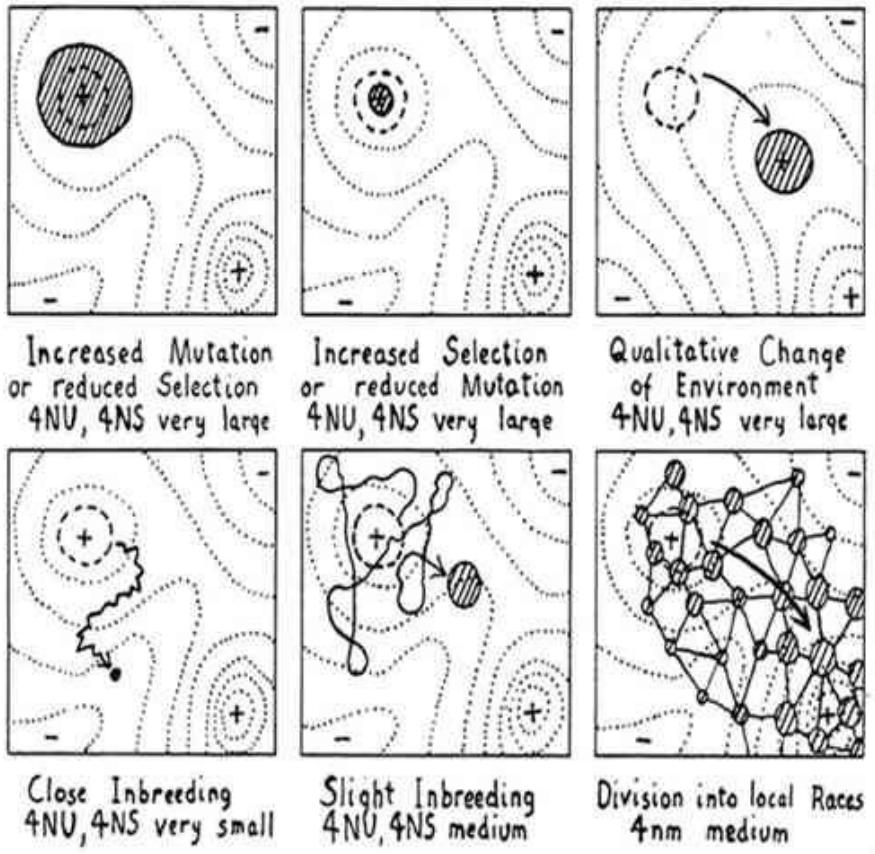
- "The notion of user cannot be precisely defined and therefore has no place in CS or SE"
  - Edsger Dijkstra, ICSE 4, 1979

- Mathematically:
  - "User"= a force that shapes goal space
- Engineering = surfing goal space
- SE = building + using software that surfs
  - Tim Menzies, FSE 2018

The shape of goal space? Surfing?







Brand's favorite landscape, 1932 Rabbits.

Complaint: "But that's not SE" Reply "Times change"

- Once upon a time, …
  - SE was not about users (Dijkstra, 79)
  - SE was not about testing (cleanroom, Mills, 1985)
  - SE was not about requirements (Paulk, 1993)
    - "Analysis and allocation of the system requirements is NOT the responsibility of the SE group but is a prerequisite for their work"
  - SE was not about deployment (before CI)
  - SE was not about AI

- But AI software is still software
  - needs maintenance, validation, interfacing, usability additions
  - i.e. AI needs software engineers