Complexity

Complexity

What kinds of problems do we mean when we talk about complexity?

- Can crowds help you write something?
  Bernstein et al. [1], Kim et al. [7], and Nebeling et al. [11]

Complexity

What kinds of problems do we mean when we talk about complexity?

- Can crowds help you write something?
  Bernstein et al. [1], Kim et al. [7], and Nebeling et al. [11]
- Can crowds critique novel designs?
  Yuan et al. [12] and Fuge et al. [2]

Complexity

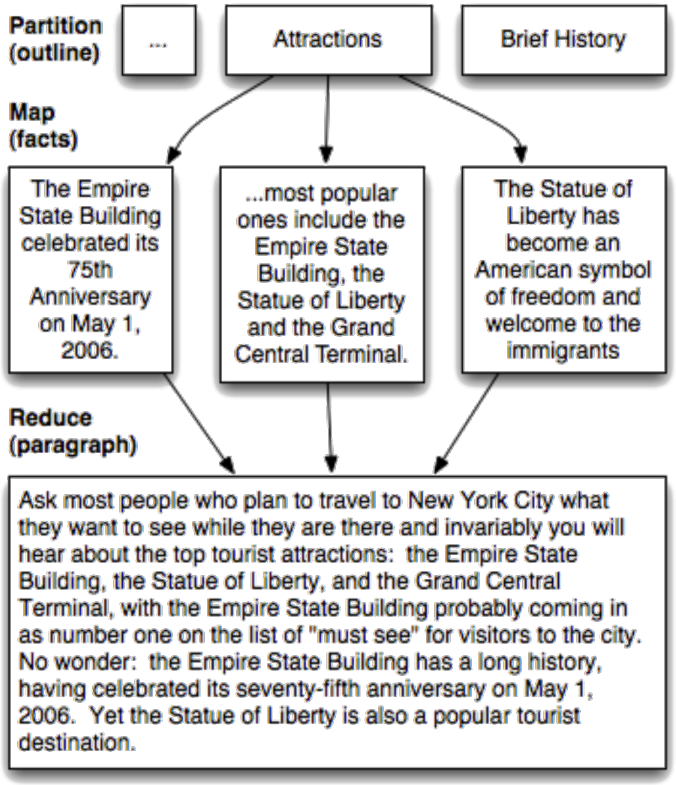
What kinds of problems do we mean when we talk about complexity?

- Can crowds help you write something?
  Bernstein et al. [1], Kim et al. [7], and Nebeling et al. [11]
- Can crowds critique novel designs?
  Yuan et al. [12] and Fuge et al. [2]
- Can crowds create things from whole cloth?

  Kim and Monroy-Hernández [5], Kim et al. [6], Hahn et al. [4], and Lasecki, Kushalnagar, and Bigham [9]

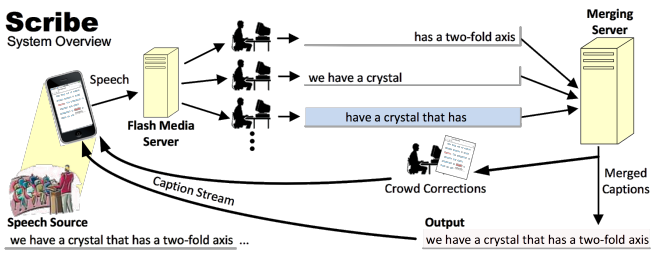
What does the crowdsourcing literature say?

- Build complexity into the process
  - **Apply CS methods to people**
    Kittur et al. [8]



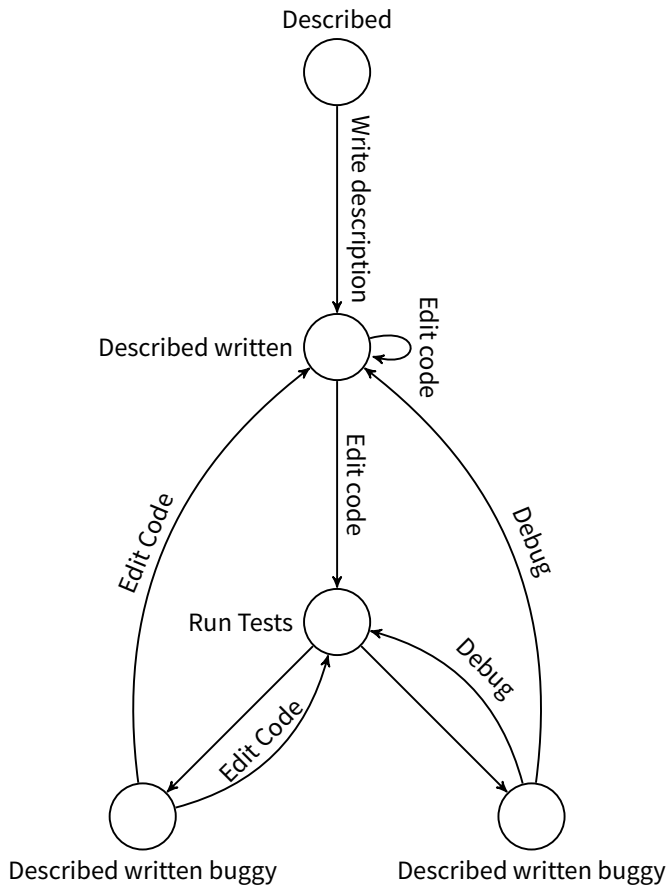
What does the crowdsourcing literature say?

- Build complexity into the process
  - Apply CS methods to people
    Kittur et al. [8]
  - **Humans as computational units**
    Lasecki, Kushalnagar, and Bigham [9]



What does the crowdsourcing literature say?

- Build complexity into the process
  - Apply CS methods to people
    Kittur et al. [8]
  - Humans as computational units
    Lasecki, Kushalnagar, and Bigham [9]
  - **Crowdsourcing workflows as function state machines**
    LaToza et al. [10]



What does the piecework literature say?

George Airy (astronomer) used a very similar approach [3]

- Employed computers
- 13–20 years old
- Overworked
- Underpaid
- Could be fired at will

George Airy — whiz kid

Airy built complexity into the process, assigning *human computers* to compute, verify, and correct the right ascension and declination of stars.

No. of Swing.	Approximate Time (Astronomical Reckoning).		Number of Signals.	Mean of Times by Shelton.	Mean of Times by Earnshaw.	Interval by Shelton.	Interval by Earnshaw.	Rate $\frac{\text{Earnshaw}}{\text{Shelton}}$	Logarithm of Rate $\frac{\text{Earnshaw}}{\text{Shelton}}$	Corrected Logarithm of Rate $\frac{\text{Earnshaw}}{\text{Shelton}}$
	Oct.	h		h m s	h m s	h m s	h m s			
1	1.	23	22	3 19 36·505	21 23 28·764	4 0 23·100	4 0 38·722	1·0010831	0·00047012	
2	2.	3	21	7 19 59·605	1 24 7·486	3 58 21·652	3 58 37·400	1·0011011	0·00047793	
3	2.	7	21	11 18 21·257	5 22 44·886	4 45 27·829	4 45 46·421	1·0010855	0·00047117	0·00047387
4	2.	11	29	16 3 49·086	10 8 31·307	4 17 6·532	4 17 23·234	1·0010827	0·00046995	
5	2.	16	17	20 20 55·618	14 25 54·541	3 13 21·898	3 13 34·795	1·0011116	0·00048249	
6	2.	19	25	23 34 17·516	17 39 29·336	3 49 42·503	3 49 57·654	1·0010994	0·00047720	0·00047990
7	2.	23	31	3 24 0·019	21 29 26·990	3 55 2·071	3 55 17·433	1·0010893	0·00047282	
8	3.	3	21	7 19 2·090	1 24 44·423	4 2 41·510	4 2 57·445	1·0010944	0·00047503	
9	3.	7	25	11 21 43·600	5 27 41·868	4 31 5·786	4 31 23·591	1·0010947	0·00047516	0·00046316
10	3.	11	22	15 52 49·386	9 59 5·459	3 27 49·747	3 28 3·324	1·0010888	0·00047260	
11	3.	15	24	19 20 39·133	13 27 8·783	3 59 47·292	4 0 3·188	1·0011049	0·00047959	
12	3.	19	24	23 20 26·425	17 27 11·971	4 3 30·416	4 3 46·029	1·0010686	0·00046384	0·00047194

Cottage Industry

First appearances of piecework:

Cottage Industry

First appearances of piecework: farms

Cottage Industry

First appearances of piecework: farms, textiles

Cottage Industry

First appearances of piecework: farms, textiles, and matchsticks.

Planes, trains, and automobiles

. . . not in that order

Trains

- Train engineers were among the first skilled workers to experience piece–rate payment systems
- “Efficiency experts” occasionally came by to evaluate how long it took to do various jobs
- This led to slow–walking the job, to offset the fact that efficiency experts weren’t accounting for liminal work.

Planes, trains, and automobiles

... not in that order

Automobiles

- Ford, Taylor, and others moved quickly monitor and measure professionals.

- *Manufacturing* proved amenable to assembly line processes.

Planes, trains, and automobiles

... not in that order

Planes

Planes, trains, and automobiles

... not in that order

Trains

Automobiles

Planes

Ford

Comparisons

References


Michael S. Bernstein et al. “Soylent: A Word Processor with a Crowd Inside”. In: *Proceedings of the 23Nd Annual ACM Symposium on User Interface Software and Technology*. UIST ’10. New York, New York, USA: ACM, 2010, pp. 313–322. ISBN: 978-1-4503-0271-5. DOI: 10.1145/1866029.1866078. URL: http://doi.acm.org/10.1145/1866029.1866078.


Mark Fuge et al. “Analysis of collaborative design networks: A case study of openideo”. In: *Journal of Computing and Information Science in Engineering* 14.2 (2014), p. 021009.


David Alan Grier. *When computers were human*. Princeton University Press, 2013.


Nathan Hahn et al. “The Knowledge Accelerator: Big Picture Thinking in Small Pieces”. In: *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. CHI ’16.

ACM, 2016, pp. 2258–2270. ISBN: 978-1-4503-3362-7. DOI: 10.1145/2858036.2858364. URL: http://doi.acm.org/10.1145/2858036.2858364.

Joy Kim and Andrés Monroy-Hernández. “Storia: Summarizing Social Media Content Based on Narrative Theory Using Crowdsourcing”. In: *Proceedings of the 19th ACM Conference on Computer–Supported Cooperative Work & Social Computing*. CSCW ’16. ACM, 2016, pp. 1018–1027. ISBN: 978-1-4503-3592-8. DOI: 10.1145/2818048.2820072. URL: http://doi.acm.org/10.1145/2818048.2820072.

Joy Kim et al. “Mechanical Novel: Crowdsourcing Complex Work through Revision”. In: *Proceedings of the 20th ACM Conference on Computer Supported Cooperative Work \& Social Computing*. 2017.

Juho Kim et al. “Crowdsourcing Step-by-step Information Extraction to Enhance Existing How-to Videos”. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI ’14. Toronto, Ontario, Canada: ACM, 2014, pp. 4017–4026. ISBN: 978-1-4503-2473-1. DOI: 10.1145/2556288.2556986. URL: http://doi.acm.org/10.1145/2556288.2556986.

Aniket Kittur et al. “CrowdForge: Crowdsourcing Complex Work”. In: *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology*. UIST ’11. ACM, 2011, pp. 43–52. ISBN: 978–1-4503–0716–1. DOI: 10.1145/2047196.2047202. URL: http://doi.acm.org/10.1145/2047196.2047202.

Walter S. Lasecki, Raja Kushalnagar, and Jeffrey P. Bigham. “Legion Scribe: Real-time Captioning by Non-experts”. In:

Proceedings of the 16th International ACM SIGACCESS Conference on Computers & Accessibility. ASSETS '14. Rochester, New York, USA: ACM, 2014, pp. 303–304. ISBN: 978-1-4503-2720-6. DOI: 10.1145/2661334.2661352. URL: http://doi.acm.org/10.1145/2661334.2661352.

Thomas D. LaToza et al. "Microtask Programming: Building Software with a Crowd". In: *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology*. UIST '14. Honolulu, Hawaii, USA: ACM, 2014, pp. 43–54. ISBN: 978-1-4503-3069-5. DOI: 10.1145/2642918.2647349. URL: http://doi.acm.org/10.1145/2642918.2647349.

Michael Nebeling et al. "WearWrite: Crowd–Assisted Writing from Smartwatches". In: *Proceedings of the 2016 CHI*

Conference on Human Factors in Computing Systems. CHI '16. ACM, 2016, pp. 3834–3846. ISBN: 978-1-4503-3362-7. DOI: 10.1145/2858036.2858169. URL: http://doi.acm.org/10.1145/2858036.2858169.

Alvin Yuan et al. "Almost an Expert: The Effects of Rubrics and Expertise on Perceived Value of Crowdsourced Design Critiques". In: *Proceedings of the 19th ACM Conference on Computer-Supported Cooperative Work & Social Computing*. CSCW '16. ACM, 2016, pp. 1005–1017. ISBN: 978-1-4503-3592-8. DOI: 10.1145/2818048.2819953. URL: http://doi.acm.org/10.1145/2818048.2819953.