

Human Computation

Core Research Questions and State of the Art

part I

Edith Law

Carnegie Mellon University

Human Computation

in a nutshell

Human Computation in a nutshell

“Some problems are hard, even for the most sophisticated AI algorithms.”

Human Computation in a nutshell

“Some problems are hard, even for the most sophisticated AI algorithms.”

“Let humans solve it ...”

Human Computation
you are a human computer

Human Computation

you are a human computer

The Norwich line steamboat train, from New-London for Boston, this **morning** ran off the track seven miles north of New-London.

morning

morning overlooks

Type the two words:



Human Computation is an **old** idea.

The Past

a very brief history

1800

1850

1900

1950

“When Computers were Human” by David A. Grier

The Past

a very brief history

Halley Comet
1750's



source: the Yerkes Observatory

1800

1850

1900

1950

“When Computers were Human” by David A. Grier

The Past

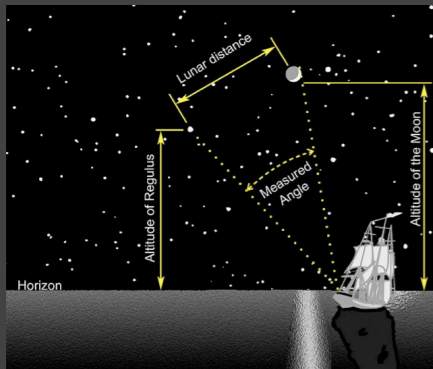
a very brief history

Halley Comet
1750's



source: the Yerkes Observatory

Nautical Almanac
1770's



1800

1850

1900

1950

“When Computers were Human” by David A. Grier

The Past

a very brief history

Halley Comet
1750's



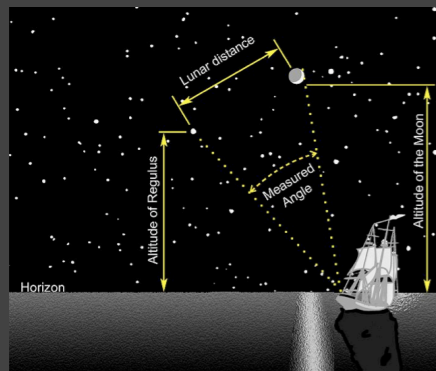
source: the Yerkes Observatory

Math Tables
Project
1930's



courtesy: David A. Grier

Nautical Almanac
1770's



1800

1850

1900

1950

“When Computers were Human” by David A. Grier

The Past

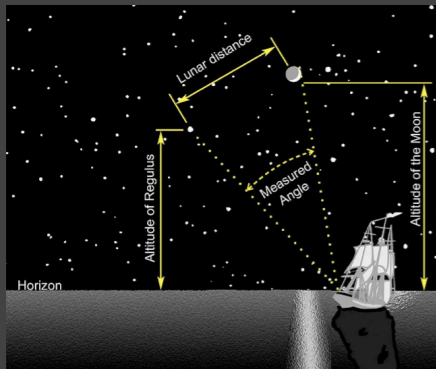
a very brief history

Halley Comet
1750's

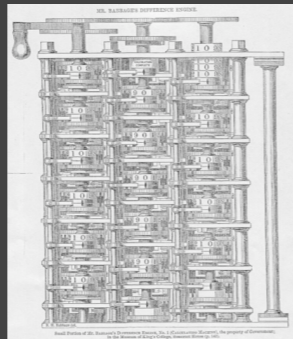


source: the Yerkes Observatory

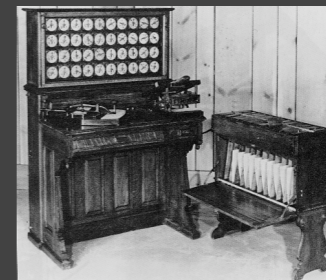
Nautical Almanac
1770's



Babbage
Difference Machine
1820's



Hollerith
Machine
1890's



source: IBM

Math Tables
Project
1930's



courtesy: David A. Grier

1800

1850

1900

1950

“When Computers were Human” by David A. Grier

The Past

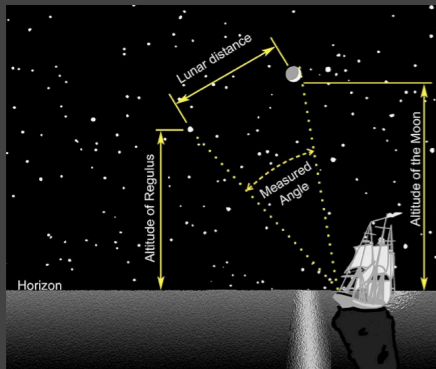
a very brief history

Halley Comet
1750's

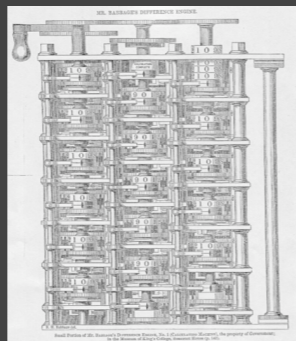


source: the Yerkes Observatory

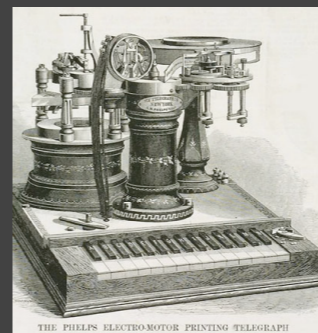
Nautical Almanac
1770's



Babbage
Difference Machine
1820's



Telegraph &
Weather
1850's



Hollerith
Machine
1890's



source: IBM

Math Tables
Project
1930's



courtesy: David A. Grier

1800

1850

1900

1950

“When Computers were Human” by David A. Grier

The **Web** changed everything.

The Present scale

score
0

 **ESP Game**
Concentrate...

time
2:46

What do you see?

taboo words
dog

guesses



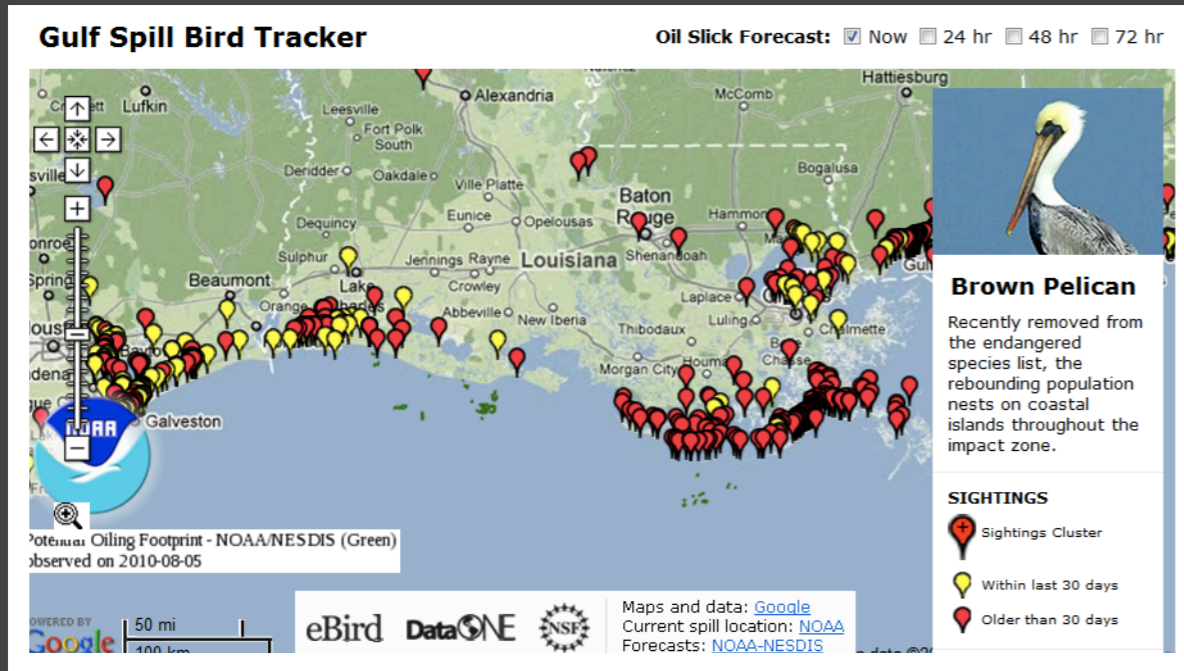
 Play Anonymously

(von Ahn and Dabbish, 2004)

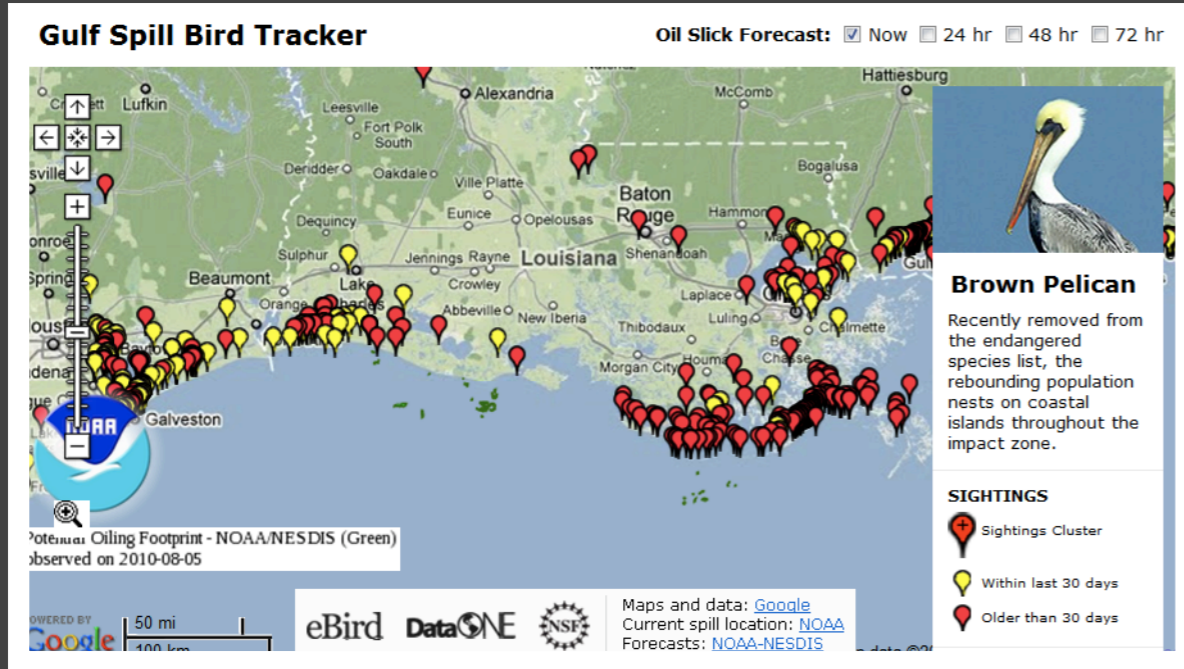
The Present

reach

The Present reach



The Present reach



ZOO NIVERSE

REAL SCIENCE ONLINE

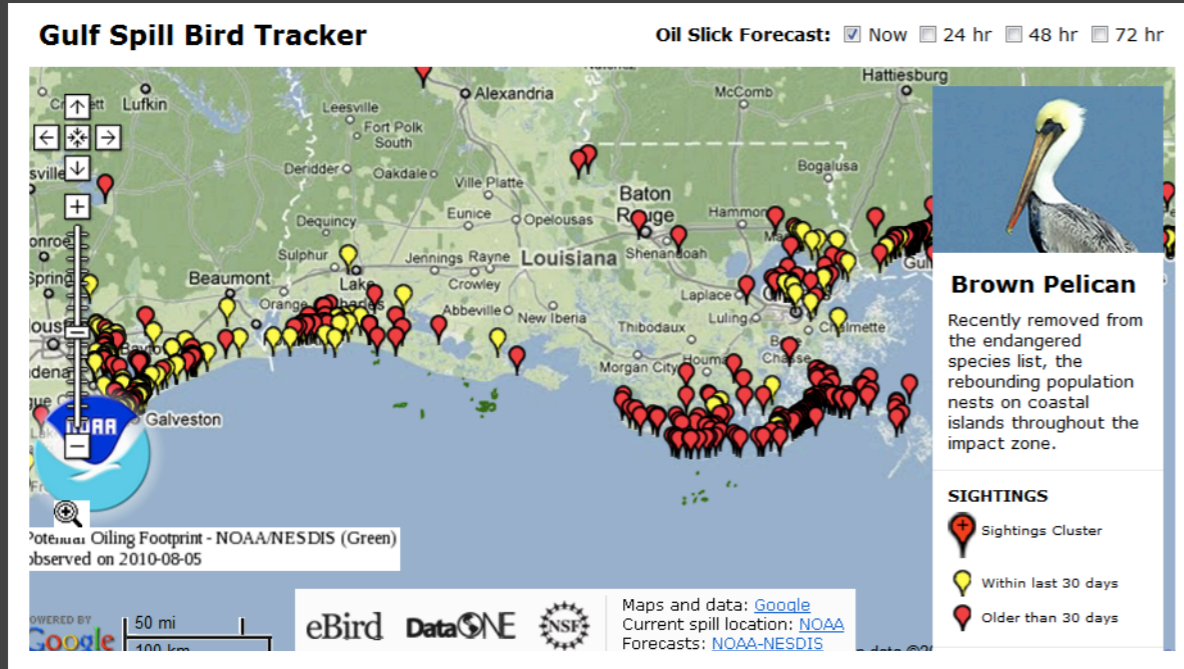
planethunters.org

GALAXY ZOO HUBBLE

SOLAR STORMWATCH

MOON ZOO

The Present reach



ZOO NIVERSE

REAL SCIENCE ONLINE

planethunters.org

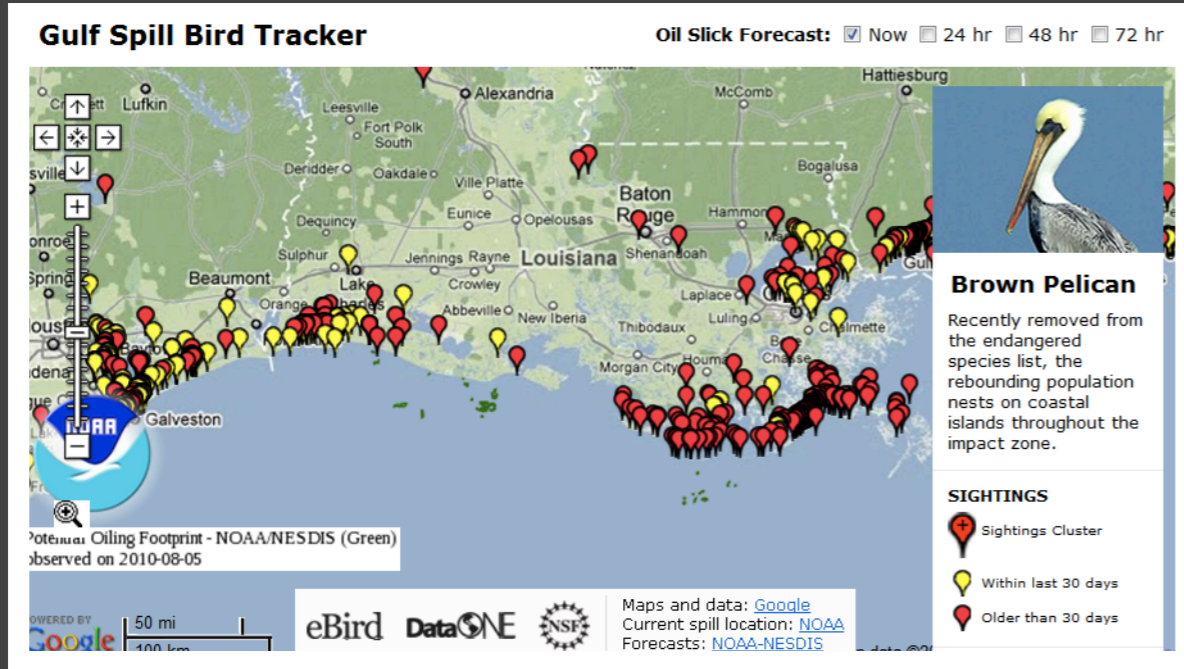
GALAXY ZOO HUBBLE

SOLAR STORMWATCH

MOON ZOO



The Present reach



ZOO NIVERSE

REAL SCIENCE ONLINE

planethunters.org

GALAXY ZOO HUBBLE

SOLAR STORMWATCH

MOON ZOO



The Great Sunflower Project

Welcome to 2010!

We are so glad that you are joining us! Please collect data as a sample from the early part of each month and the later flowers. We also changed the protocol. You do not have to count you simply record all the bees in 15 minutes (rather than 30 minutes). We're hoping that will make it easier! Before sampling, please note some of the factors that we think influence the density of bees to sample:

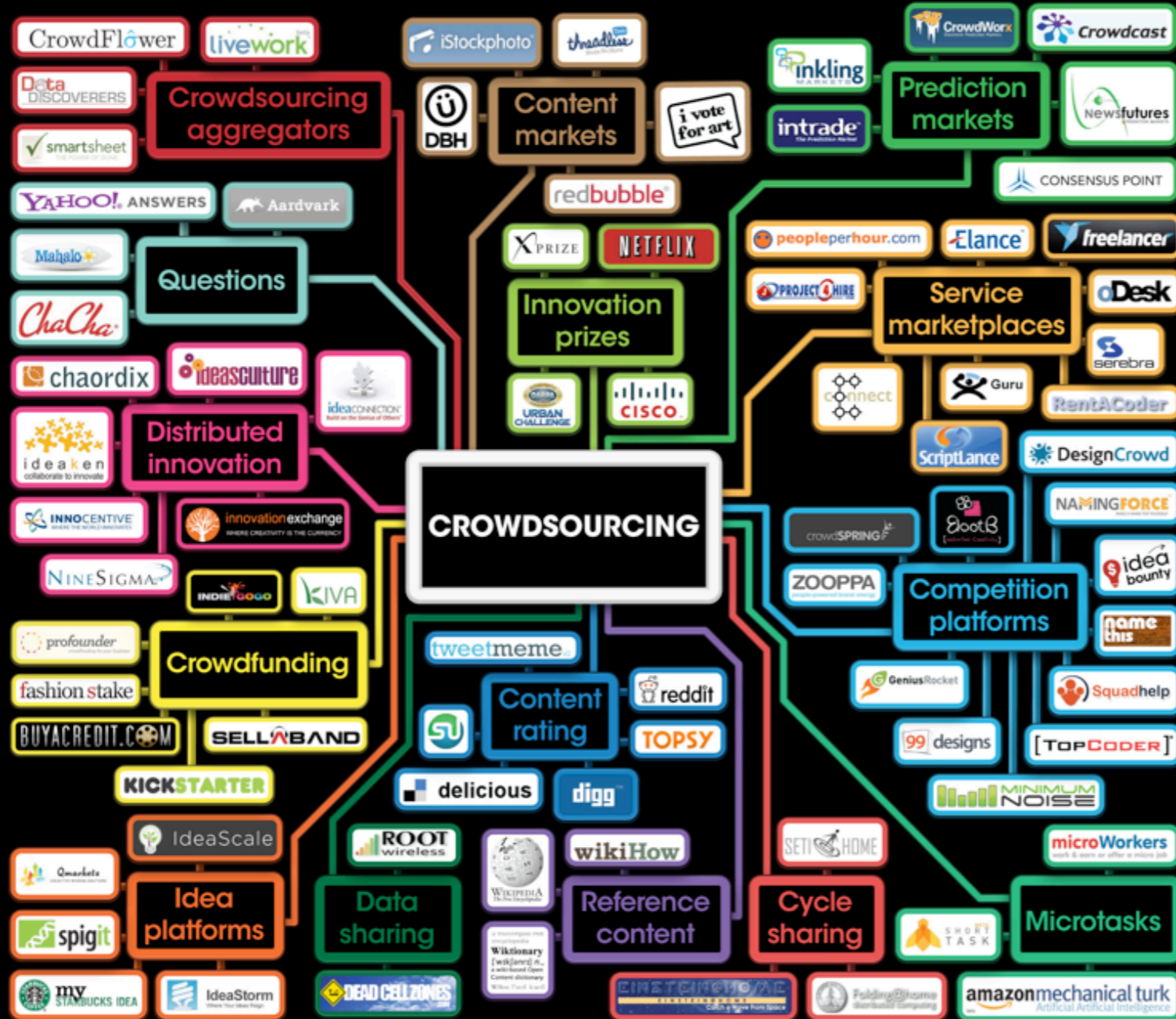
Join the Hunt for Bees

The Great Sunflower Project
www.greatsunflower.org
sfbee@sfsu.edu

Sunflowers
Helianthus annuus

The Present pervasiveness

CROWDSOURCING LANDSCAPE Beta v1



Common Crowdsourcing Tasks and Examples	
3D object design	Thingiverse
Advertising	Idea bounty
Business ideas	Ushahidi
Clothing	thecreative
Consumer research	ClickAdvisor
Crisis information	Ushahidi
Data analysis	SETI@HOME
Fact checking	PolitiFact.com
Graphic design	99 designs
Human reading	oDesk
Investigative reporting	theguardian
Journalism	seed
Lending	zoopa
Mapping	99 designs
Movie reviews	IMDb
Music	musiquest
Observation	GALAXY ZOO
Patent research	Ushahidi
Philanthropy	SETI@HOME
Political activism	MoveOn.org
Product design	MUJI
Proofreading	oDesk
Scientific problems	fold
Software	SETI@HOME
Software development	RentACoder
Software testing	oTest
Stock picking	Marketocracy
Tagging	Google
Translation	facebook
Trends	TREND-HUNTER
TV programming	current
Word of mouth	BzzAgent
Writing and editing	oDesk

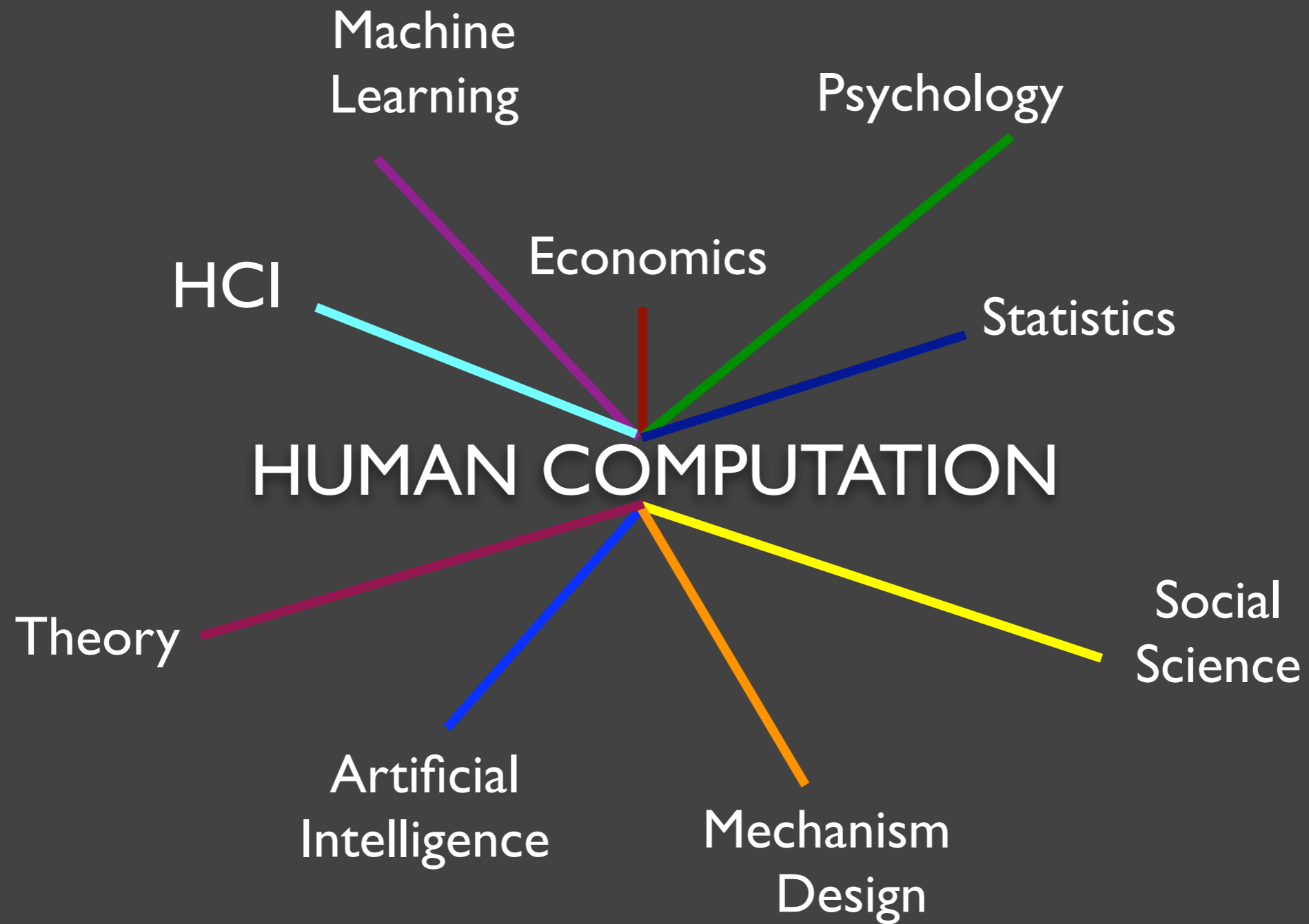
Human Computation

a growing field

1st Human Computation Workshop	KDD 2009	
Crowdsourcing for Search Evaluation	SIGIR 2010	
2nd Human Computation Workshop	KDD 2010	
Advancing Computer Vision with Humans in the Loop	CVPR 2010	W
Creating Speech and Language Data with Amazon's Mechanical Turk	NAACL 2010	O
Computational Social Science and Wisdom of the Crowds	NIPS 2010	R
Workshop on Ubiquitous Crowdsourcing	UbiComp 2010	K
Enterprise Crowdsourcing Workshop	ICWE 2010	S
Collaborative Translation Technology, Crowdsourcing and the Translator	AMTA 2010	H
Crowdsourcing for Search and Data Mining	WSDM 2010	O
Workshop on Crowdsourcing for Information Retrieval	SIGIR 2011	P
Workshop on Social Computing and User Generated Content	EC 2011	S
Workshop on Crowdsourcing and Human Computation	CHI 2011	
3rd Human Computation Workshop	AAAI 2011	
Mechanical Turk for Computer Vision	CVPR 2010	T
Crowdsourcing for Relevance Evaluation	ECIR 2010	U
Managing Crowdsourced Human Computation	WWW 2011	T
Crowdsourcing 101: Putting the WSDM of Crowds to Work for You	WSDM 2011	O
Crowdsourcing Applications and Platforms	VLDB 2011	R
Crowdsourcing for Information Retrieval: : Principles, Methods and Applications	SIGIR 2011	I
Quality Crowdsourcing for Human Computer Interaction Research	HCIC 2011	A
Crowdsourcing for Fun and Profit	CrowdConf 2011	L
Human Computation: Core Research Questions and State of the Art	AAAI 2011	S

Human Computation

multi-disciplinary



Tutorial

with a purpose

Introduce a framework for human computation with a set of **concepts**, core **research questions**, **existing work** and **open problems**.

Tutorial

with a purpose

Introduce a framework for human computation with a set of **concepts**, core **research questions**, **existing work** and **open problems**.

Research Opportunities indicated by 

Tutorial

a rough schedule

ALGORITHM

part 1

2:00-3:30
Edith Law

DESIGN

part 2

4:00-5:30
Luis von Ahn

I

A FRAMEWORK FOR HUMAN COMPUTATION

Concepts • Scope

CONCEPTS

Computation

a general definition

The process of mapping **input** to **output**.

Computational Problems

examples

multiplication	two numbers	➔	product
sorting	set of objects	➔	set of objects sorted
medical diagnosis	x-ray, lab tests	➔	diagnosis
object recognition	image	➔	tag
translation	source sentence	➔	target sentence
editing	text	➔	corrected text
planning	goal, constraints	➔	sequence of actions

Human Computation

a general definition

Computation that is carried out by a **human**.

Human Computation problem statement

Human Computation problem statement

Given a computational problem, design a solution using human computers and automated computers.

Related Concepts

definitions

Related Concepts definitions

COLLECTIVE INTELLIGENCE

The shared or **group intelligence** that emerges from the collaboration and competition of many individuals (bacteria, animals, humans, computer agents).

Related Concepts

definitions

COLLECTIVE INTELLIGENCE

The shared or **group intelligence** that emerges from the collaboration and competition of many individuals (bacteria, animals, humans, computer agents).

SOCIAL COMPUTING

Technology for supporting **social behavior and interactions** (e.g., blog, email, Instant messaging) or **group computation** (e.g., collaborative filtering, auctions, prediction markets).

Related Concepts

definitions

COLLECTIVE INTELLIGENCE

The shared or **group intelligence** that emerges from the collaboration and competition of many individuals (bacteria, animals, humans, computer agents).

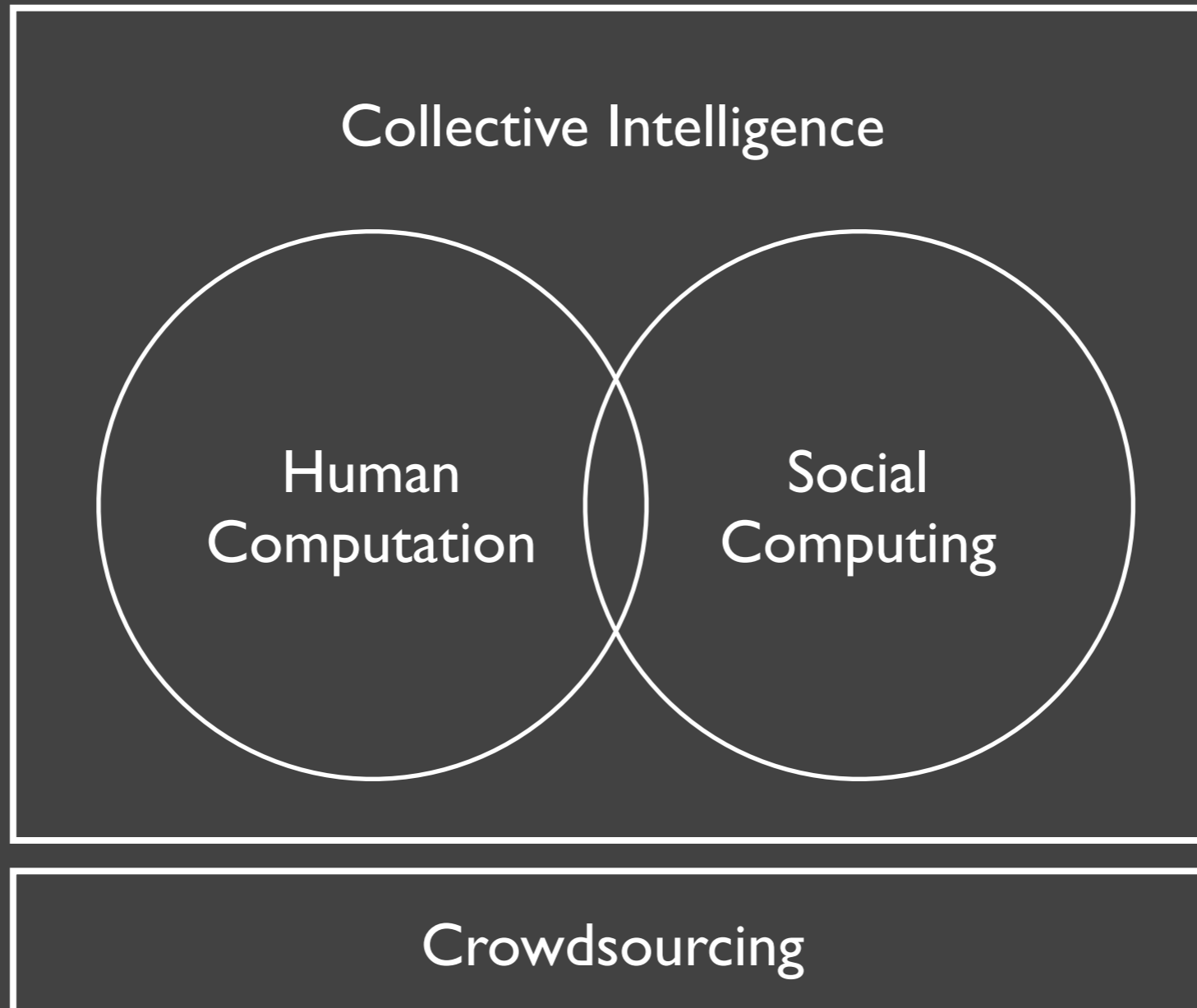
SOCIAL COMPUTING

Technology for supporting **social behavior and interactions** (e.g., blog, email, Instant messaging) or **group computation** (e.g., collaborative filtering, auctions, prediction markets).

CROWDSOURCING

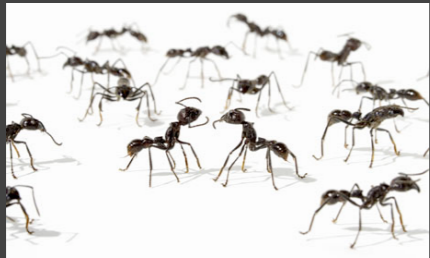
Outsourcing tasks through an **open call**.

Related Concepts boundaries

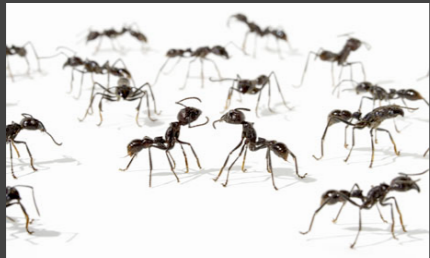


Three Distinguishing Features of human computation

Three Distinguishing Features of human computation

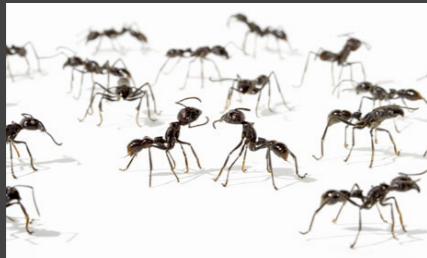


Three Distinguishing Features of human computation



“Human” In The Loop
not bacteria, not ants, not fish.

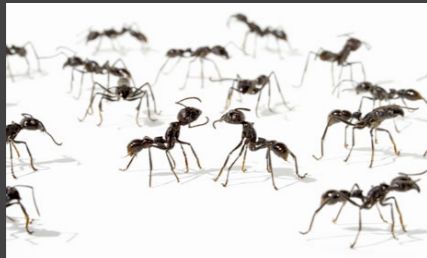
Three Distinguishing Features of human computation



“Human” In The Loop
not bacteria, not ants, not fish.



Three Distinguishing Features of human computation



“Human” In The Loop

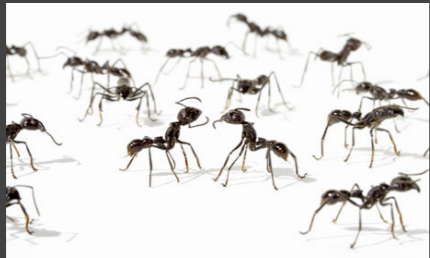
not bacteria, not ants, not fish.



Conscious Effort

humans are actively computing something, not merely carrier of sensors and computational devices.

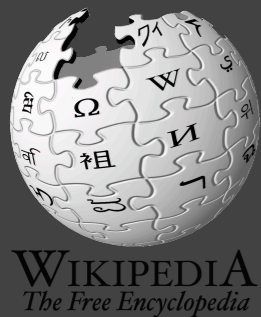
Three Distinguishing Features of human computation



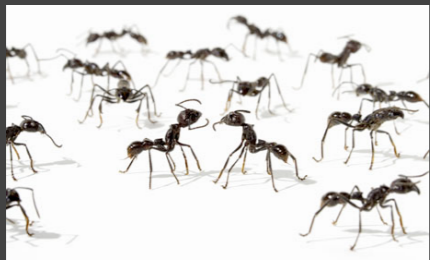
“Human” In The Loop
not bacteria, not ants, not fish.



Conscious Effort
humans are actively computing something, not merely carrier of sensors and computational devices.



Three Distinguishing Features of human computation



“Human” In The Loop
not bacteria, not ants, not fish.



Conscious Effort

humans are actively computing something, not merely carrier of sensors and computational devices.



WIKIPEDIA
The Free Encyclopedia

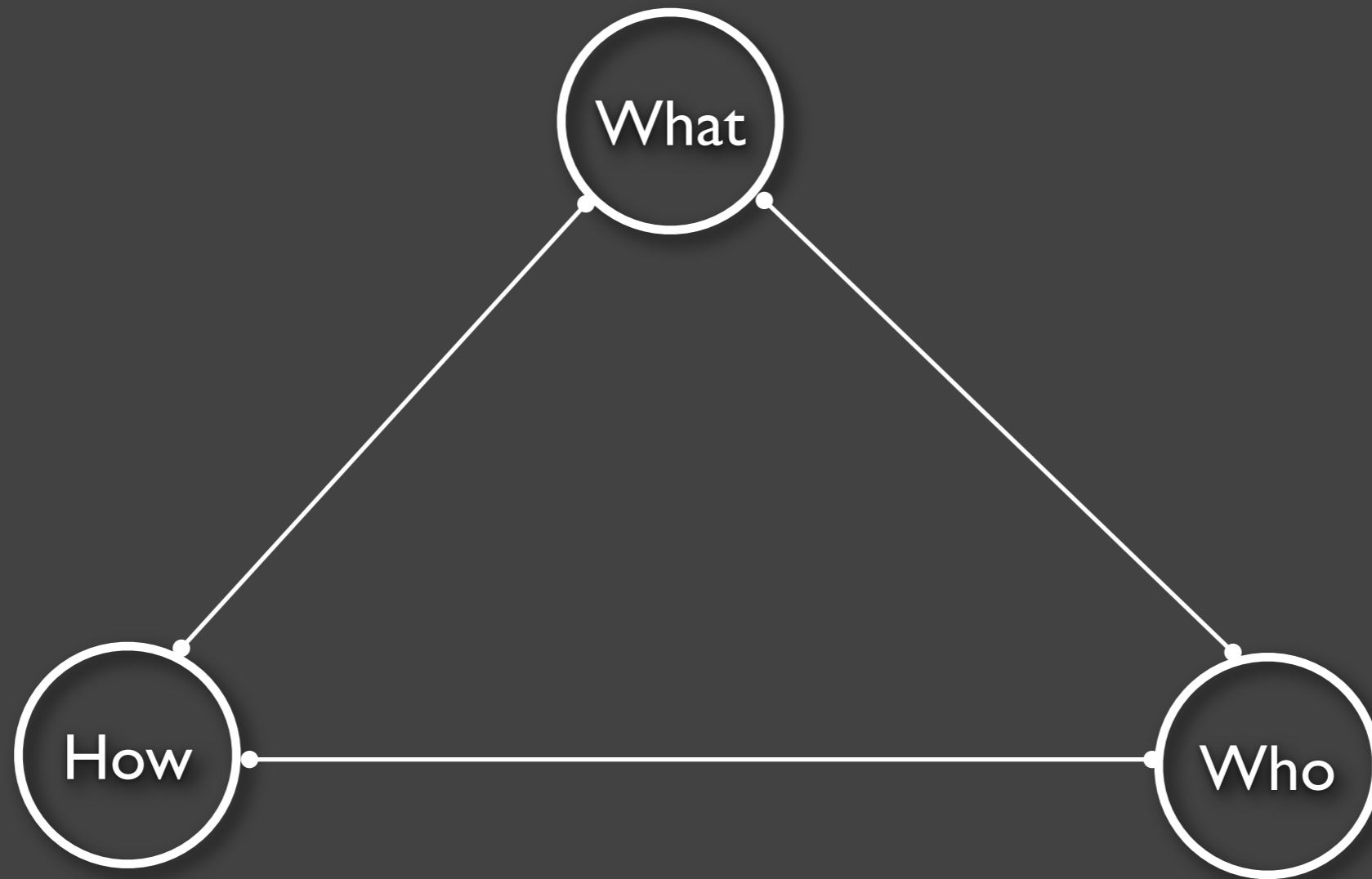
Explicit Control

the outcome of the computation is determined by an algorithm, and not the natural dynamics of the crowd.

SCOPE

Core Research Questions

“what”, “who”, “how”



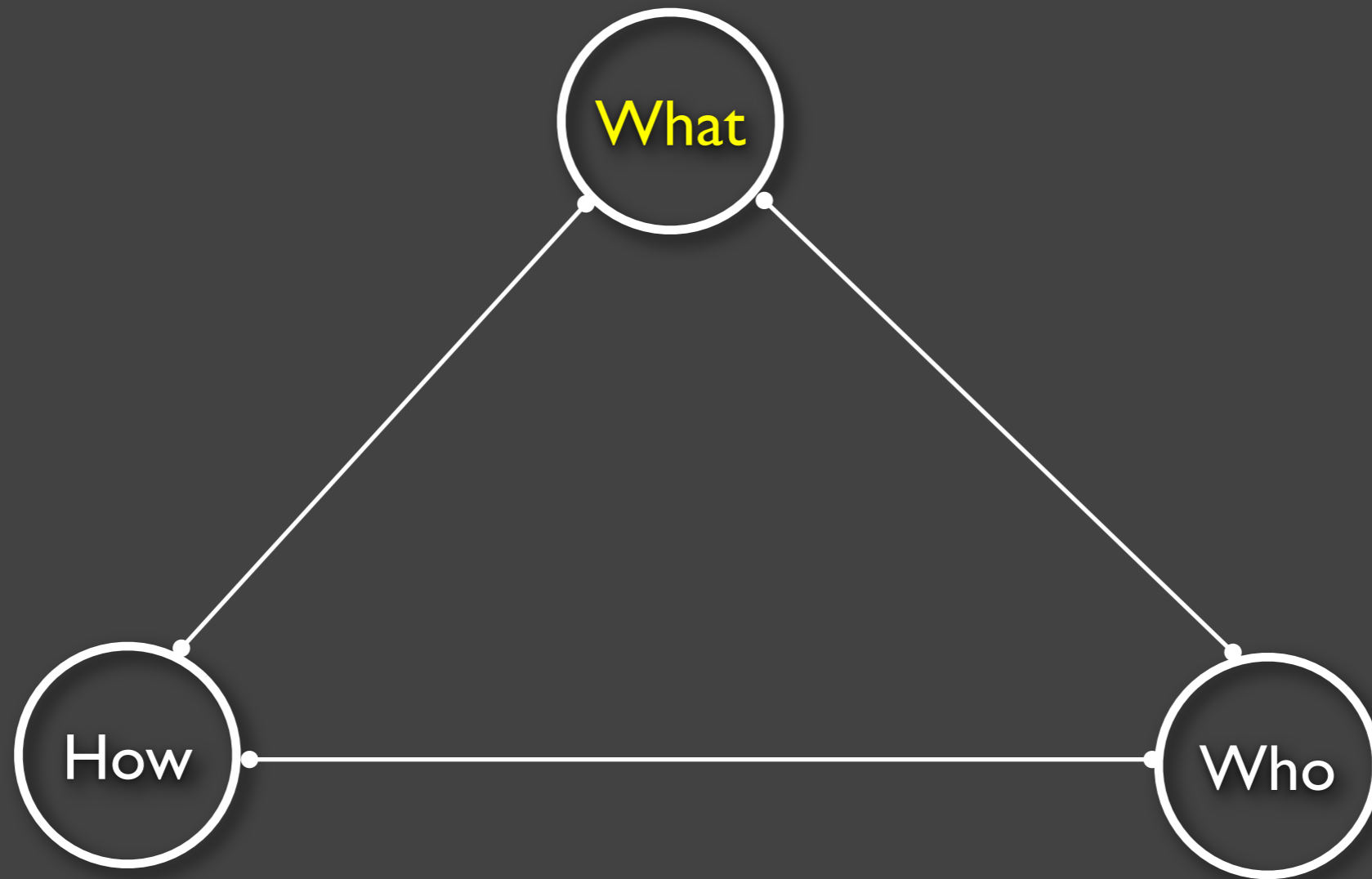
Core Research Questions

from the definition

Given a computational problem, design a solution using human computers and automated computers.

Core Research Questions

“what”, “who”, “how”



Core Research Questions

from the definition

“How hard is the problem? Is it efficiently solvable?”



Given a **computational problem**, design a solution using human computers and automated computers.

Core Research Questions

from the definition

“Is the human computation algorithm correct and efficient?”



Given a computational problem, design a **solution** using human computers and automated computers.

Core Research Questions

from the definition

Given a computational problem, design a solution using **human computers** and automated computers.



“How do we aggregate the outputs of many human computers?”

Core Research Questions

from the definition

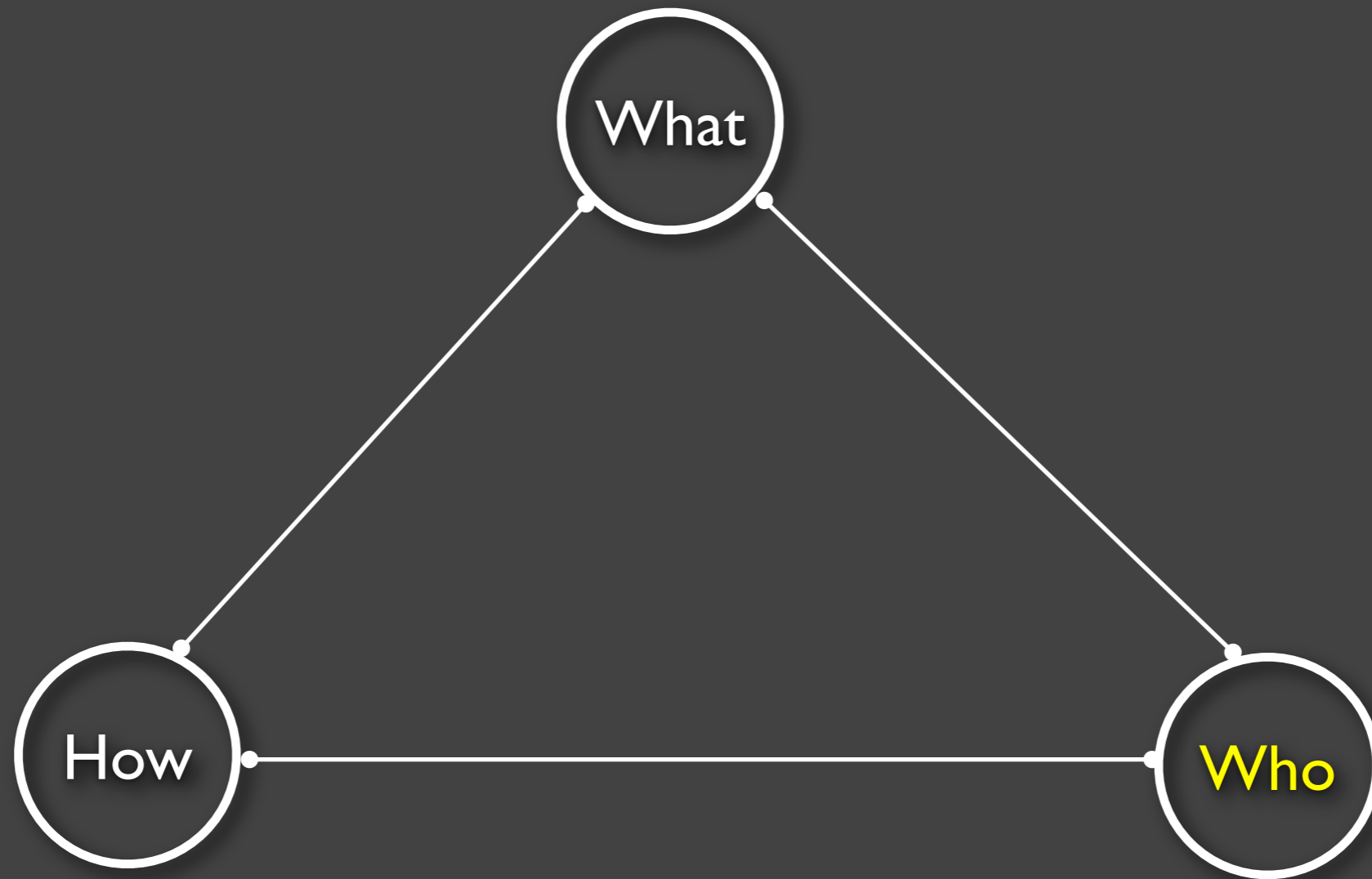
Given a computational problem, design a solution using
human computers and automated computers.



“How to make the tradeoff between human versus machine?”

Core Research Questions

“what”, “who”, “how”



Core Research Questions

from the definition

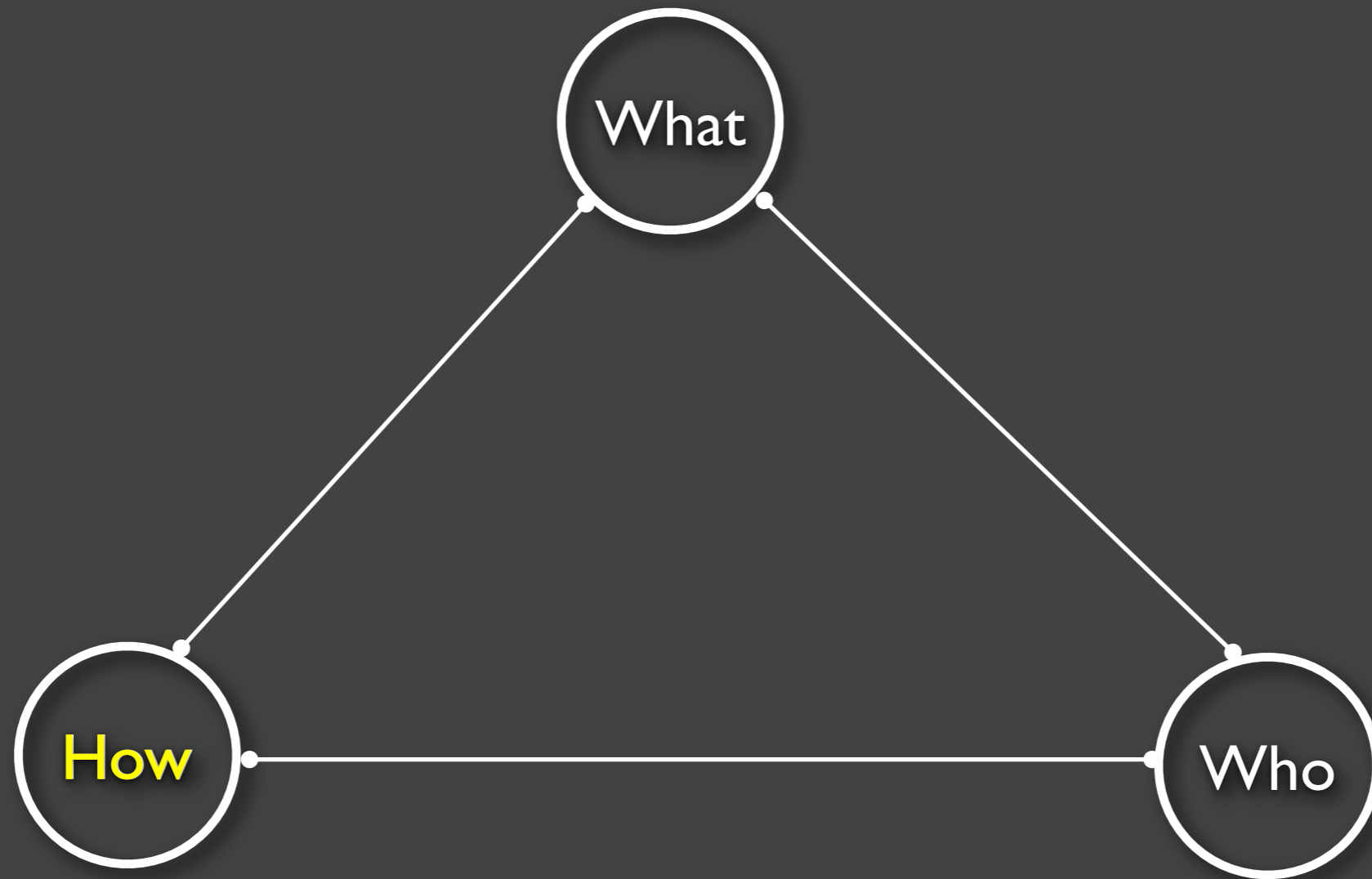
“To whom do we route each task, and how?”



Given a computational problem, design a solution **using** human computers and automated computers.

Core Research Questions

“what”, “who”, “how”



Core Research Questions

from the definition

Given a computational problem, design a solution using **human** computers and automated computers.



“How to design tasks, motivate participation and incentivize truthful outputs?”

Core Research Questions

from the definition

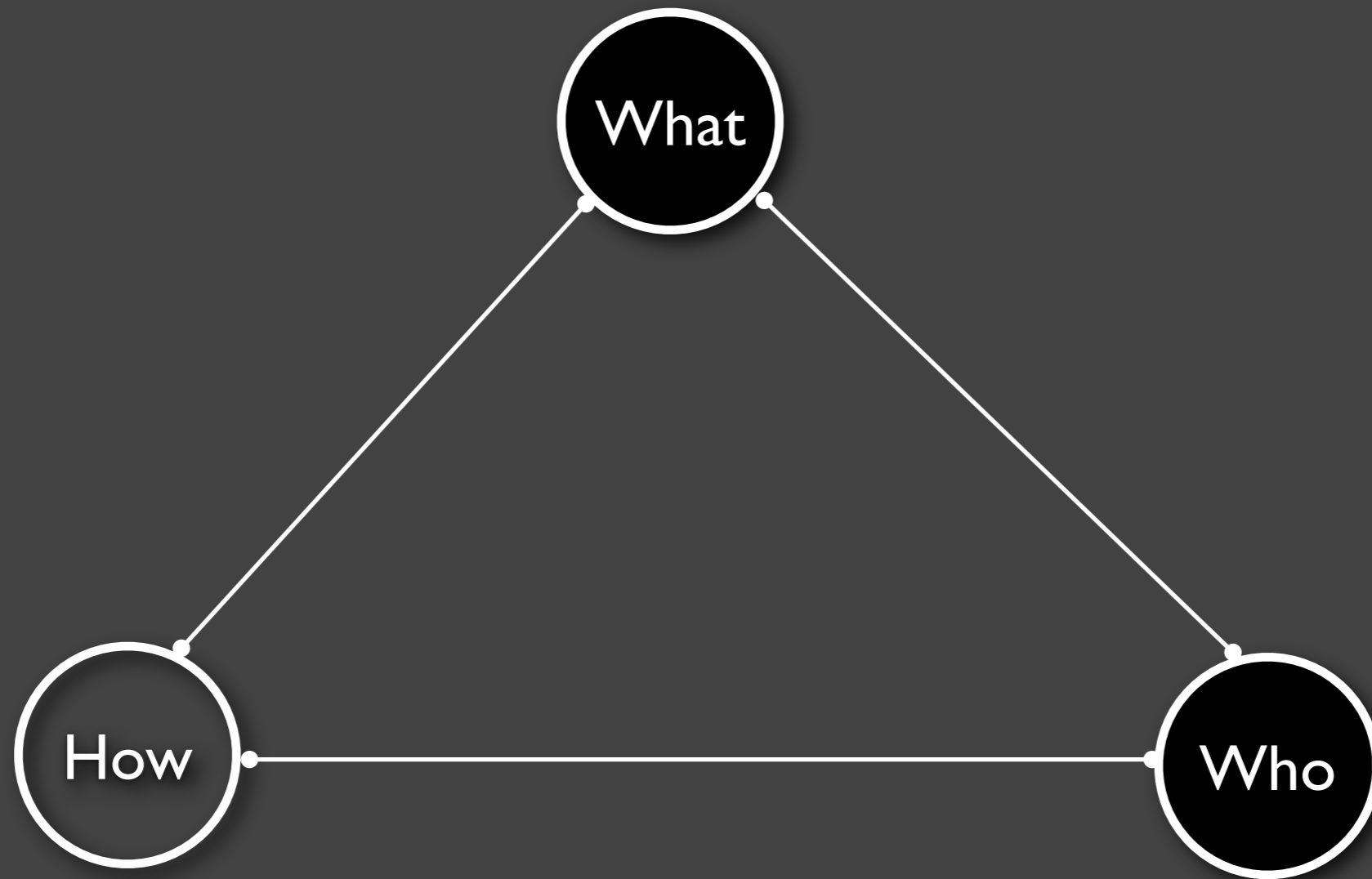
“How to meet the needs and wants of the requesters?”



Given a computational problem, design a solution using human computers and automated computers.

Core Research Questions

“what”, “who”, “how”



II

HUMAN COMPUTATION ALGORITHMS

Definition • Properties

DEFINITION

What are **algorithms**?

Algorithms

an example

```
function quicksort(A)
  initialize empty lists L and G
  if (length(A) ≤ 1)
    return A
  pivot = A.remove(find_pivot(A));
  for x in A
    if compare(x, pivot)
      L.add(x)
    else
      G.add(x)
  return concatenate(quicksort(L), pivot, quicksort(G))

function pivot(A)
  return randomIndex(A);

function compare(x, pivot)
  return (x < pivot)
```

Algorithms

an example

Inputs



```
function quicksort(A)
  initialize empty lists L and G
  if (length(A) ≤ 1)
    return A
  pivot = A.remove(find_pivot(A));
  for x in A
    if compare(x, pivot)
      L.add(x)
    else
      G.add(x)
  return concatenate(quicksort(L), pivot, quicksort(G))

function pivot(A)
  return randomIndex(A);

function compare(x, pivot)
  return (x < pivot)
```


Algorithms

an example

Inputs



```
function quicksort(A)
  initialize empty lists L and G
  if (length(A) ≤ 1)
    return A
  pivot = A.remove(find_pivot(A));
  for x in A
    if compare(x, pivot)
      L.add(x)
    else
      G.add(x)
  return concatenate(quicksort(L), pivot, quicksort(G))

function pivot(A)
  return randomIndex(A);

function compare(x, pivot)
  return (x < pivot)
```

Outputs →

Algorithms

an example

Inputs



```
function quicksort(A)
  initialize empty lists L and G
  if (length(A) ≤ 1)
    return A
  pivot = A.remove(find_pivot(A));
  for x in A
    if compare(x, pivot)
      L.add(x)
    else
      G.add(x)
  return concatenate(quicksort(L), pivot, quicksort(G))

function pivot(A)
  return randomIndex(A);

function compare(x, pivot)
  return (x < pivot)
```

Outputs →

← Precisely
Defined
Steps

Algorithms an example

Inputs



```
function quicksort(A)
  initialize empty lists L and G
  if (length(A) ≤ 1)
    return A
  pivot = A.remove(find_pivot(A));
  for x in A
    if compare(x, pivot)
      L.add(x)
    else
      G.add(x)
  return concatenate(quicksort(L), pivot, quicksort(G))

function pivot(A)
  return randomIndex(A);

function compare(x, pivot)
  return (x < pivot)
```

Termination
Condition →

Outputs →

← Precisely
Defined
Steps

Algorithms an example

Inputs



```
function quicksort(A)
  initialize empty lists L and G
  if (length(A) ≤ 1)
    return A
  pivot = A.remove(find_pivot(A));
  for x in A
    if compare(x, pivot)
      L.add(x)
    else
      G.add(x)
  return concatenate(quicksort(L), pivot, quicksort(G))

function pivot(A)
  return randomIndex(A);

function compare(x, pivot)
  return (x < pivot)
```

Termination
Condition →

Outputs →

Sufficiently
Basic →

Operations →

← Precisely
Defined
Steps

Algorithms

a definition

“An algorithm is a finite set of rules which gives a sequence of operations for solving a specific type of problem, with five important properties:

Input, Output, Finiteness, Definiteness, Effectiveness.”

- Knuth, 1973

Algorithms

a definition

“An algorithm is a finite set of rules which gives a sequence of operations for solving a specific type of problem, with five important properties:

Input, Output, Finiteness, Definiteness, Effectiveness.”

- Knuth, 1973

Algorithms

a definition

“An algorithm is a finite set of rules which gives a sequence of operations for solving a specific type of problem, with five important properties:

Input, Output, Finiteness, Definiteness, Effectiveness.”

- Knuth, 1973

Algorithms

a definition

“An algorithm is a finite set of rules which gives a sequence of operations for solving a specific type of problem, with five important properties:

Input, **Output**, Finiteness, Definiteness, Effectiveness.”

- Knuth, 1973

Algorithms

a definition

“An algorithm is a finite set of rules which gives a sequence of operations for solving a specific type of problem, with five important properties:

Input, Output, Finiteness, Definiteness, Effectiveness.”

- Knuth, 1973

Algorithms

a definition

“An algorithm is a finite set of rules which gives a sequence of operations for solving a specific type of problem, with five important properties:

Input, Output, **Finiteness**, Definiteness, Effectiveness.”

- Knuth, 1973

Algorithms

a definition

“An algorithm is a finite set of rules which gives a sequence of operations for solving a specific type of problem, with five important properties:

Input, Output, Finiteness, Definiteness, Effectiveness.”

- Knuth, 1973

Algorithms

a definition

“An algorithm is a finite set of rules which gives a sequence of operations for solving a specific type of problem, with five important properties:

Input, Output, Finiteness, **Definiteness**, Effectiveness.”

- Knuth, 1973

Algorithms

a definition

“An algorithm is a finite set of rules which gives a sequence of operations for solving a specific type of problem, with five important properties:

Input, Output, Finiteness, Definiteness, Effectiveness.”

- Knuth, 1973

Algorithms

a definition

“An algorithm is a finite set of rules which gives a sequence of operations for solving a specific type of problem, with five important properties:

Input, Output, Finiteness, Definiteness, **Effectiveness.**”

- Knuth, 1973

Algorithms

a definition

“An algorithm is a finite set of rules which gives a sequence of operations for solving a specific type of problem, with five important properties:

Input, Output, Finiteness, Definiteness, Effectiveness.”

- Knuth, 1973

What are **human computation algorithms**?

Human Computation Algorithms

human-driven operation

```
function quicksort(A)
  initialize empty lists L and G
  if (length(A) ≤ 1)
    return A
  pivot = A.remove(find_pivot(A));
  for x in A
    if compare(x, pivot)
      L.add(x)
    else
      G.add(x)
  return concatenate(quicksort(L), pivot, quicksort(G))

function pivot(A)
  return randomIndex(A);

function compare(x, pivot)
  return (x < pivot)
```

Human Computation Algorithms

human-driven operation

```
function quicksort(A)
  initialize empty lists L and G
  if (length(A) ≤ 1)
    return A
  pivot = A.remove(find_pivot(A));
  for x in A
    if compare(x, pivot)
      L.add(x)
    else
      G.add(x)
  return concatenate(quicksort(L), pivot, quicksort(G))

function pivot(A)
  return randomIndex(A);
```

Human Computation Algorithms

human-driven operation

```
function quicksort(A)
  initialize empty lists L and G
  if (length(A) ≤ 1)
    return A
  pivot = A.remove(find_pivot(A));
  for x in A
    if compare(x, pivot)
      L.add(x)
    else
      G.add(x)
  return concatenate(quicksort(L), pivot, quicksort(G))

function pivot(A)
  return randomIndex(A);

function compare(x, pivot)
  return human_compare(x, pivot)
```

Human Computation Algorithms

human-driven operation

```
function quicksort(A)
  initialize empty lists L and G
  if (length(A) ≤ 1)
    return A
  pivot = A.remove(find_pivot(A));
  for x in A
    if compare(x, pivot)
      L.add(x)
    else
      G.add(x)
  return concatenate(quicksort(L), pivot, quicksort(G))

function pivot(A)
  return randomIndex(A);

function compare(x, pivot)
  return human_compare(x, pivot)
```



Human Computation Algorithms

human-driven operation

```
function quicksort(A)
  initialize empty lists L and G
  if (length(A) ≤ 1)
    return A
  pivot = A.remove(find_pivot(A));
  for x in A
    if compare(x, pivot)
      L.add(x)
    else
      G.add(x)
  return concatenate(quicksort(L), pivot, quicksort(G))

function pivot(A)
  return randomIndex(A);

function compare(x, pivot)
  return human_compare(x, pivot)
```

Mechanical Turk Task

Instructions

You are shown two images. You must select the image that is more indicative of suspicious activities.

Task

Imagine that you are a security guard and you are monitoring two places. Someone informed you that there are suspicious activities in one of the places, but you were not told which one. Which place will you attend to?



Submit



Human Computation Algorithms

human-driven operation

```
function quicksort(A)
  initialize empty lists L and G
  if (length(A) ≤ 1)
    return A
  pivot = A.remove(find_pivot(A));
  for x in A
    if compare(x, pivot)
      L.add(x)
    else
      G.add(x)
  return concatenate(quicksort(L), pivot, quicksort(G))

function pivot(A)
  return randomIndex(A);

function compare(x, pivot)
  return human_compare(x, pivot)
```

Mechanical Turk Task

Instructions

You are shown two images. You must select the image that is more indicative of suspicious activities.

Task

Imagine that you are a security guard and you are monitoring two places. Someone informed you that there are suspicious activities in one of the places, but you were not told which one. Which place will you attend to?



Submit

TurKit (Little et al., 2010); Boto (<http://code.google.com/p/boto/>)

Human Computation Algorithms


human-driven operation

```
function quicksort(A)
  initialize empty lists L and G
  if (length(A) ≤ 1)
    return A
  pivot = A.remove(find_pivot(A));
  for x in A
    if compare(x, pivot)
      L.add(x)
    else
      G.add(x)
  return concatenate(quicksort(L), pivot, quicksort(G))

function pivot(A)
  return randomIndex(A);

function compare(x, pivot)
  return human_compare(x, pivot)
```

TurKit (Little et al., 2010); Boto (<http://code.google.com/p/boto/>)



Human Computation Algorithms

human-driven operation

```
function quicksort(A)
  initialize empty lists L and G
  if (length(A) ≤ 1)
    return A
  pivot = A.remove(find_pivot(A));
  for x in A
    if compare(x, pivot)
      L.add(x)
    else
      G.add(x)
  return concatenate(quicksort(L), pivot, quicksort(G))

function pivot(A)
  return randomIndex(A);

function compare(x, pivot)
  return human_compare(x, pivot)
```



Human Computation Algorithms

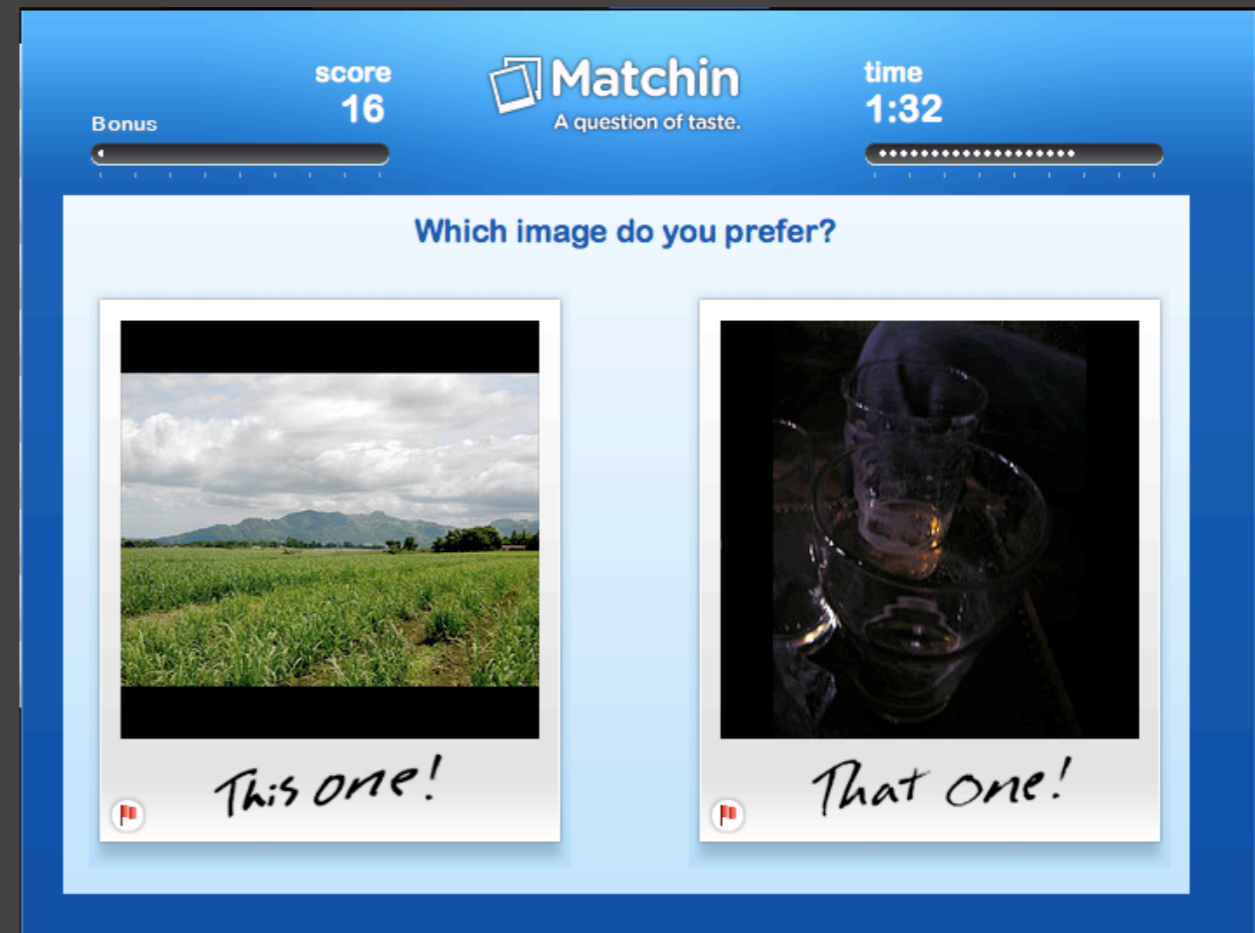
human-driven operation

```
function quicksort(A)
  initialize empty lists L and G
  if (length(A) ≤ 1)
    return A
  pivot = A.remove(find_pivot(A));
  for x in A
    if compare(x, pivot)
      L.add(x)
    else
      G.add(x)
  return concatenate(quicksort(L), pivot, quicksort(G))

function pivot(A)
  return randomIndex(A);

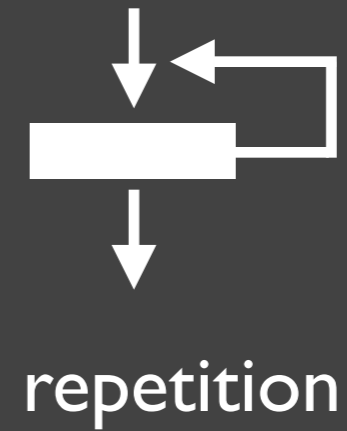
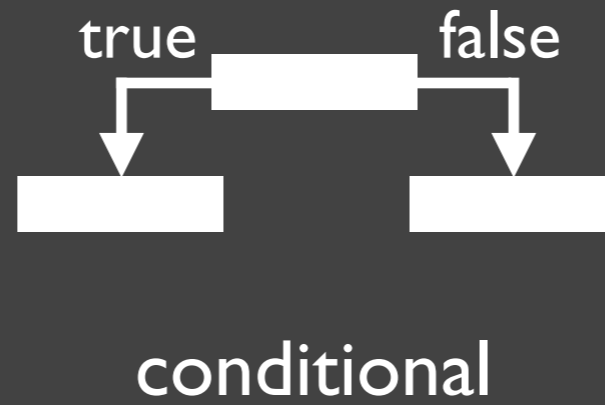
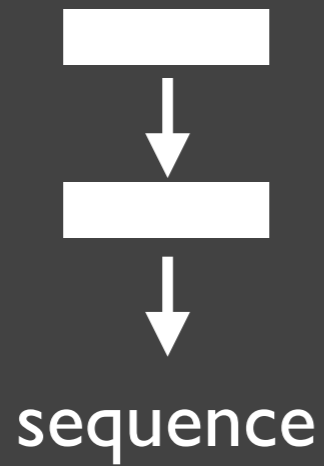
function compare(x, pivot)
  return human_compare(x, pivot)
```

Games with a Purpose



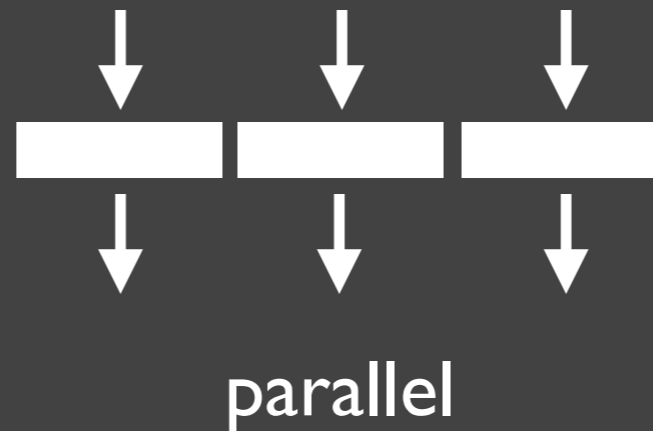
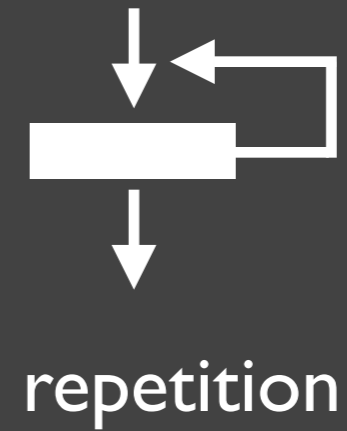
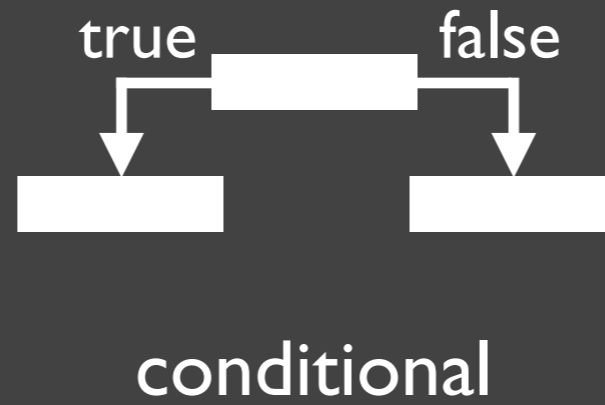
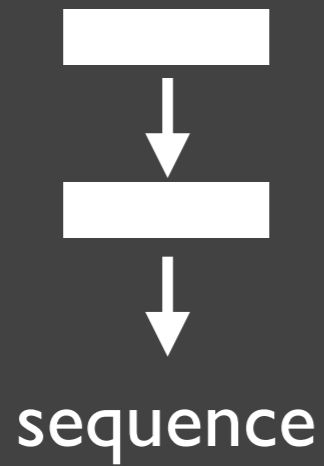
Human Computation Algorithms

human-driven controls



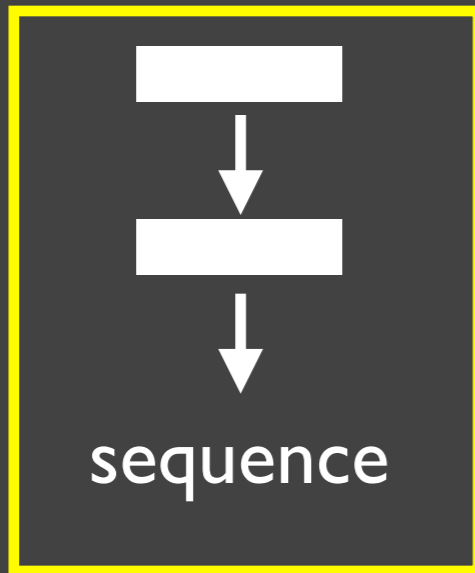
Human Computation Algorithms

human-driven controls

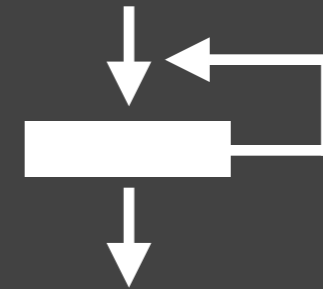


Human Computation Algorithms

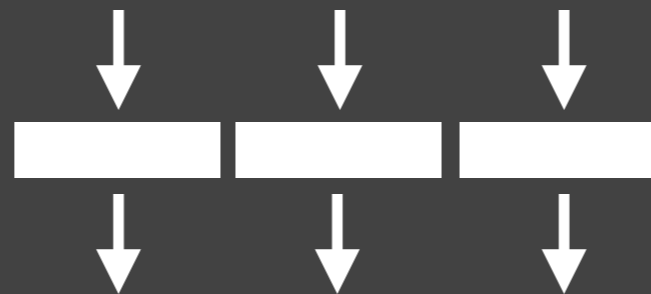
human-driven controls



conditional



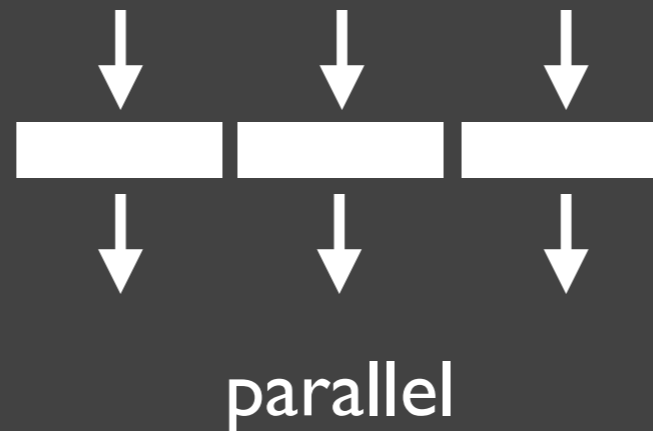
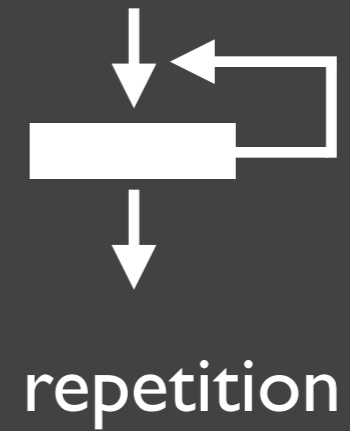
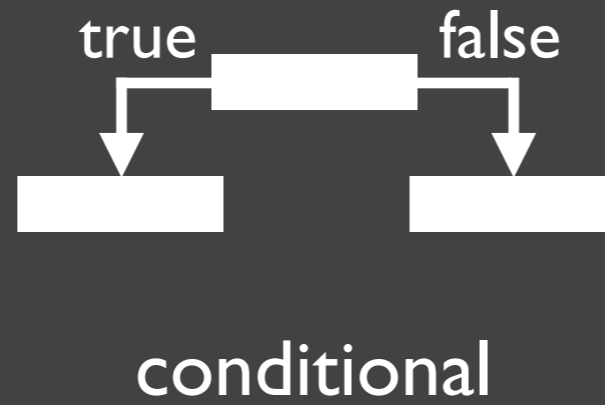
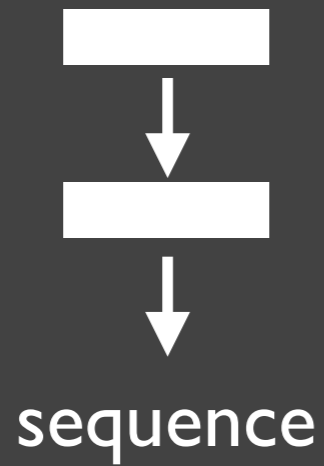
repetition



parallel

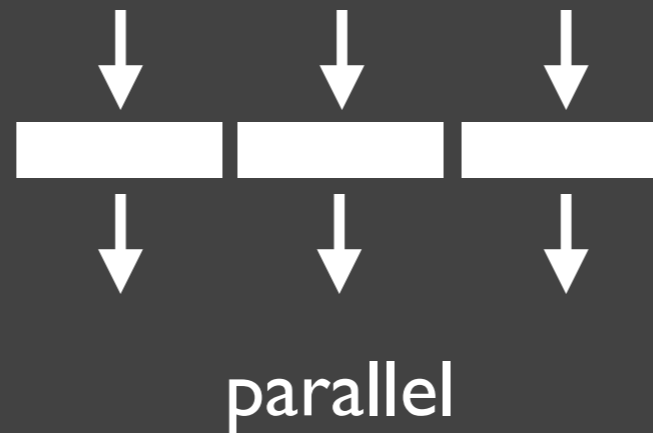
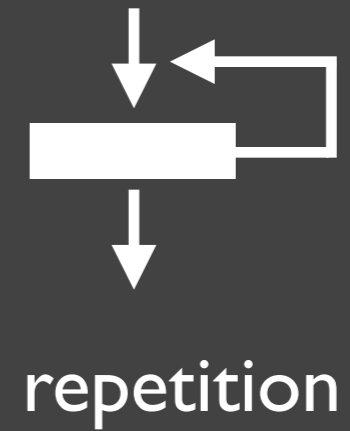
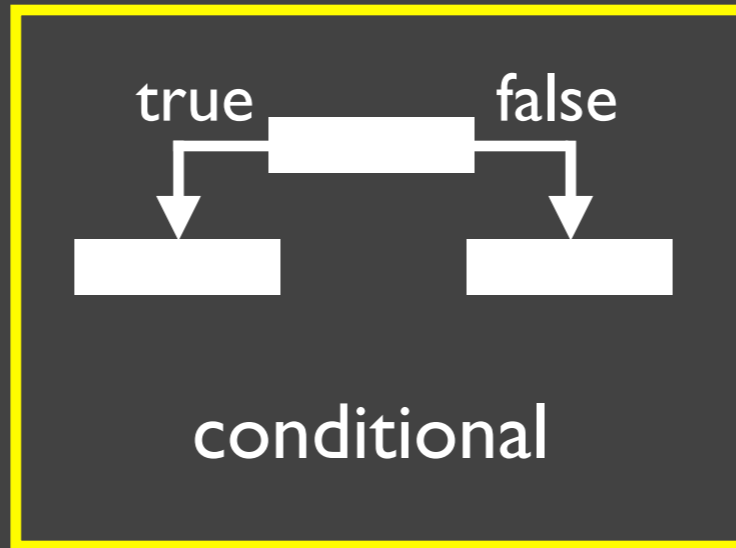
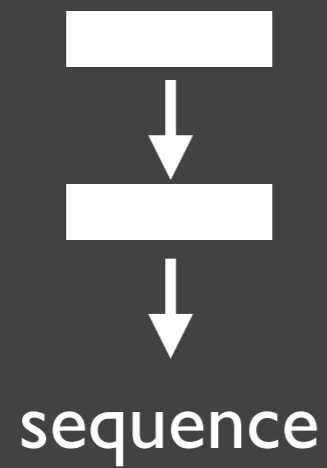
Human Computation Algorithms

human-driven controls



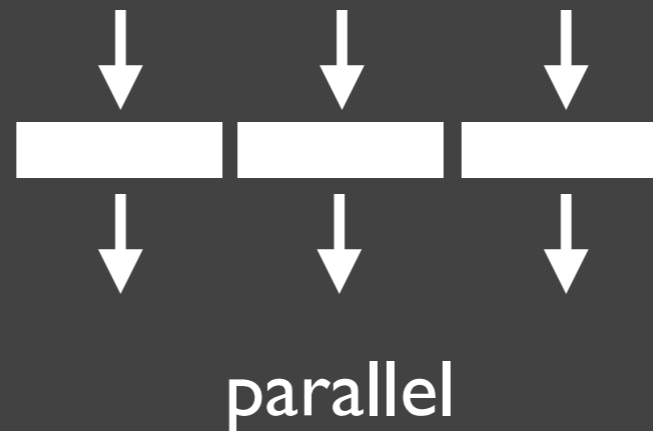
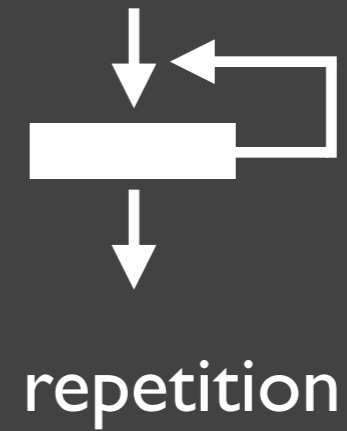
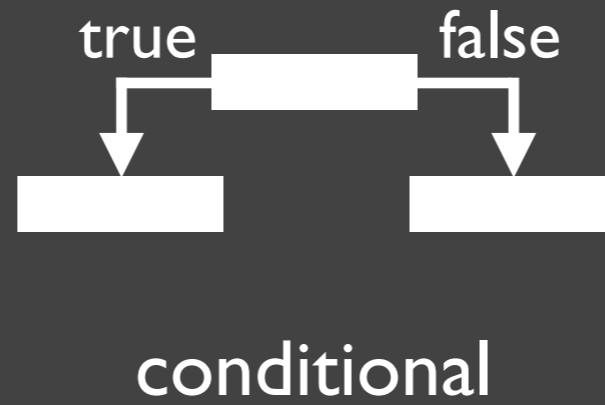
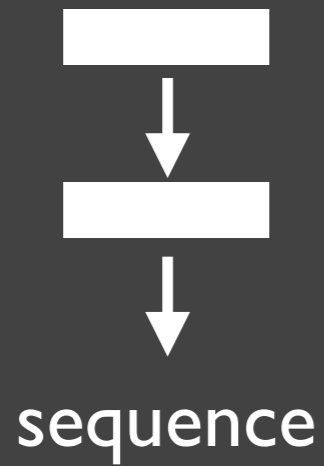
Human Computation Algorithms

human-driven controls



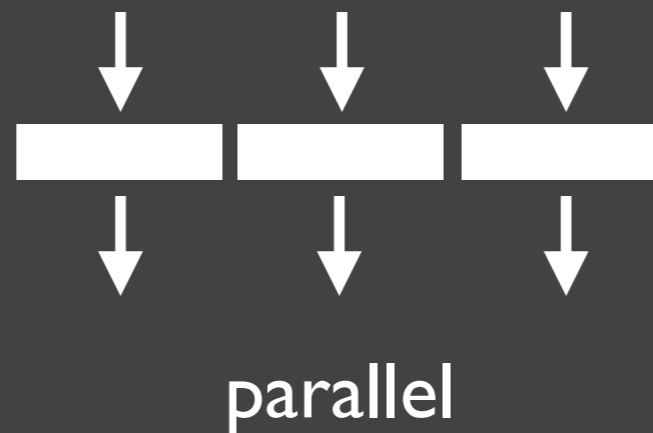
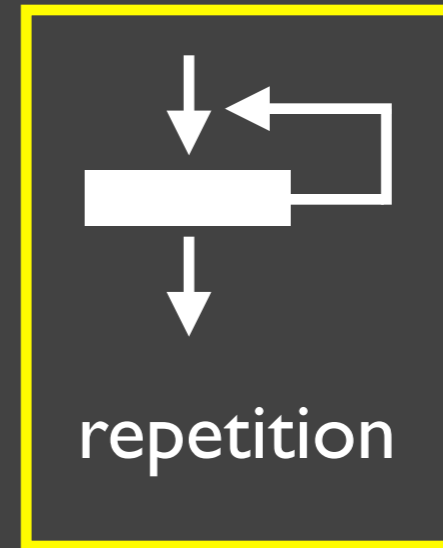
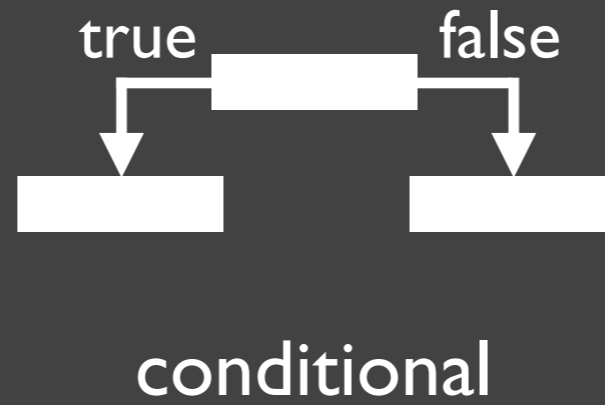
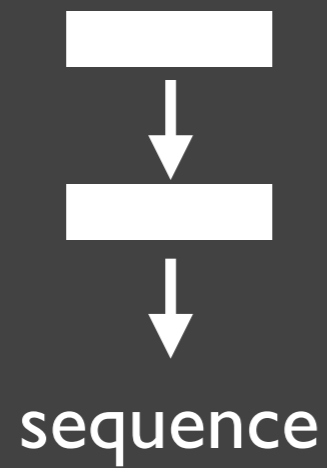
Human Computation Algorithms

human-driven controls



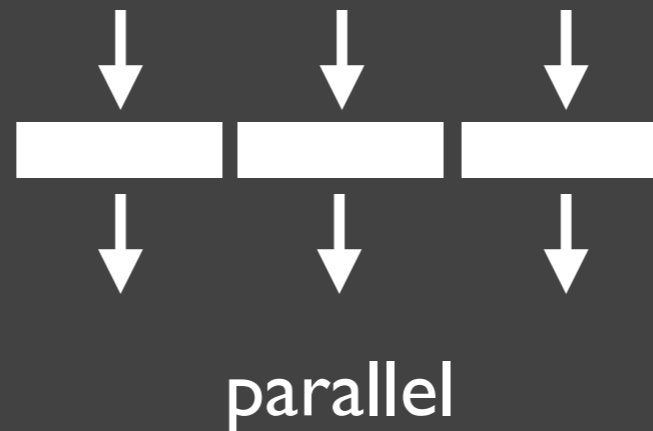
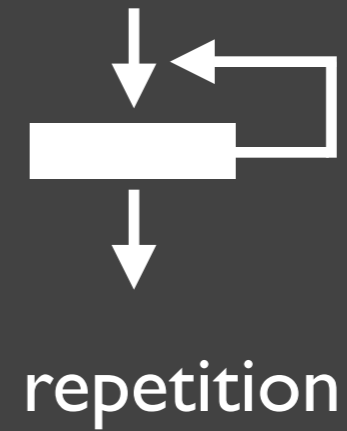
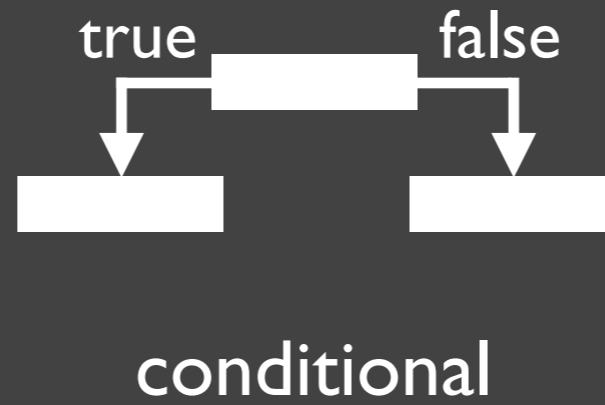
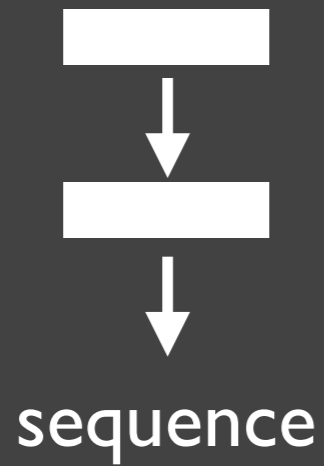
Human Computation Algorithms

human-driven controls



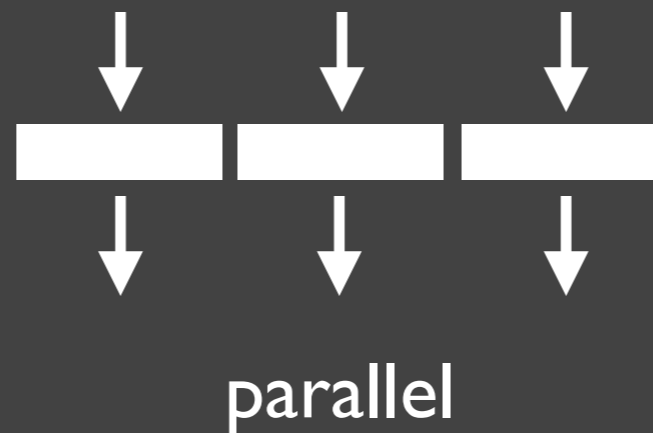
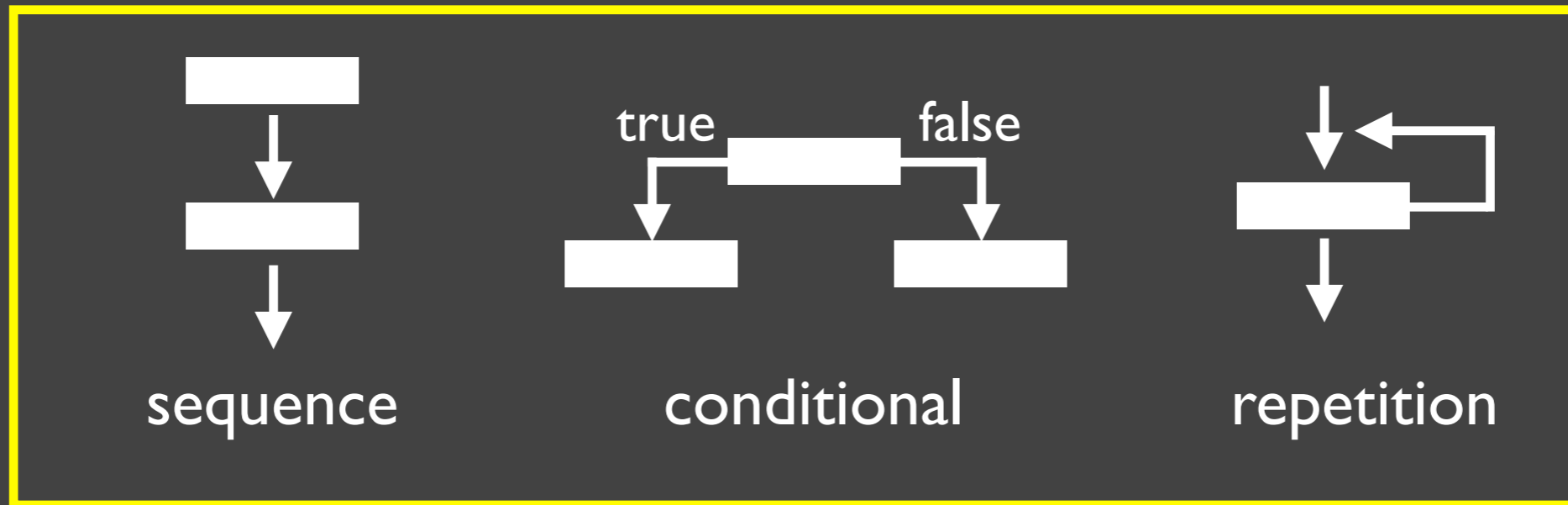
Human Computation Algorithms

human-driven controls



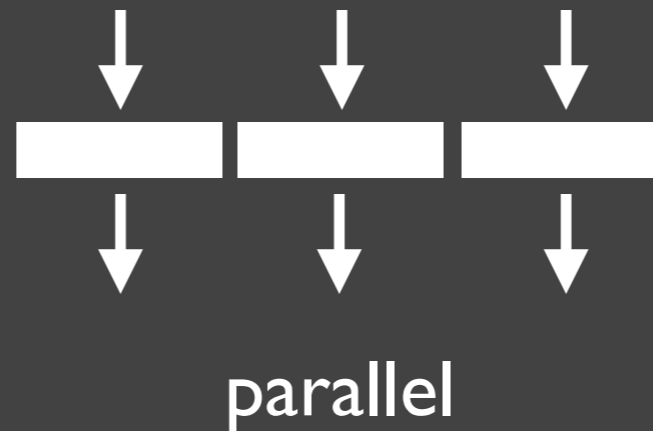
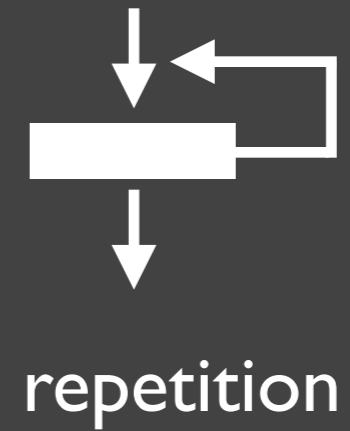
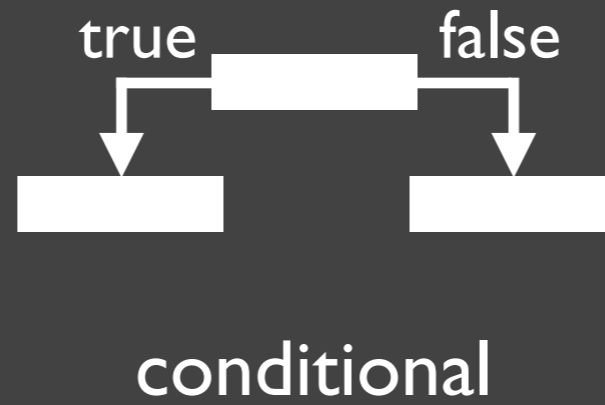
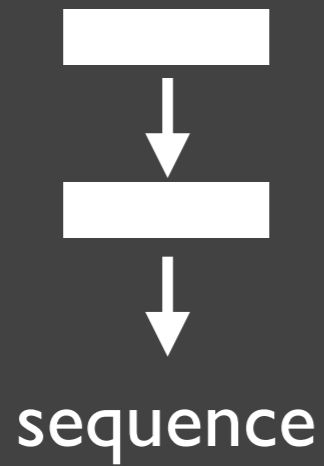
Human Computation Algorithms

human-driven controls



Human Computation Algorithms

human-driven controls



Human Computation Algorithms

human-driven program synthesis

Human Computation Algorithms

human-driven program synthesis

plan a wedding

Human Computation Algorithms

human-driven program synthesis

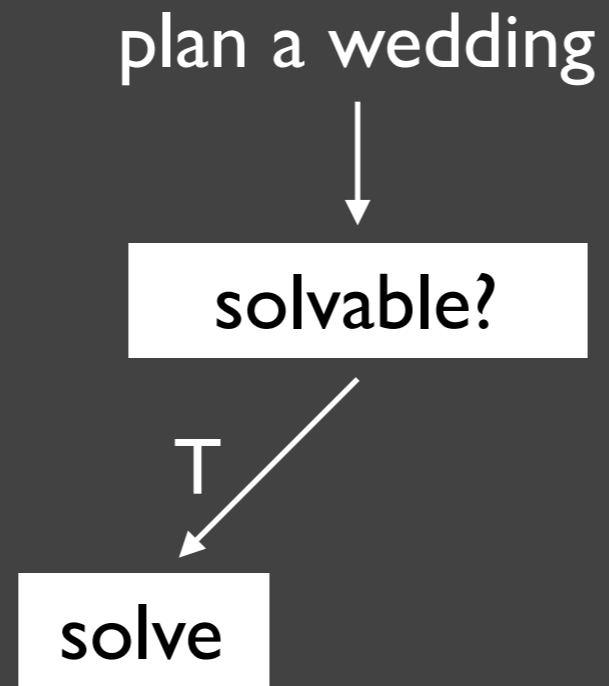
plan a wedding



solvable?

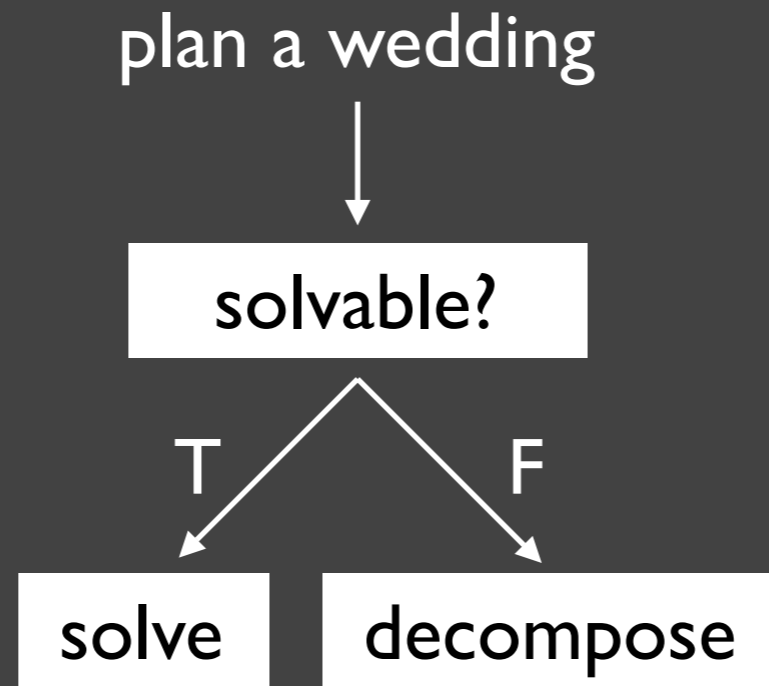
Human Computation Algorithms

human-driven program synthesis



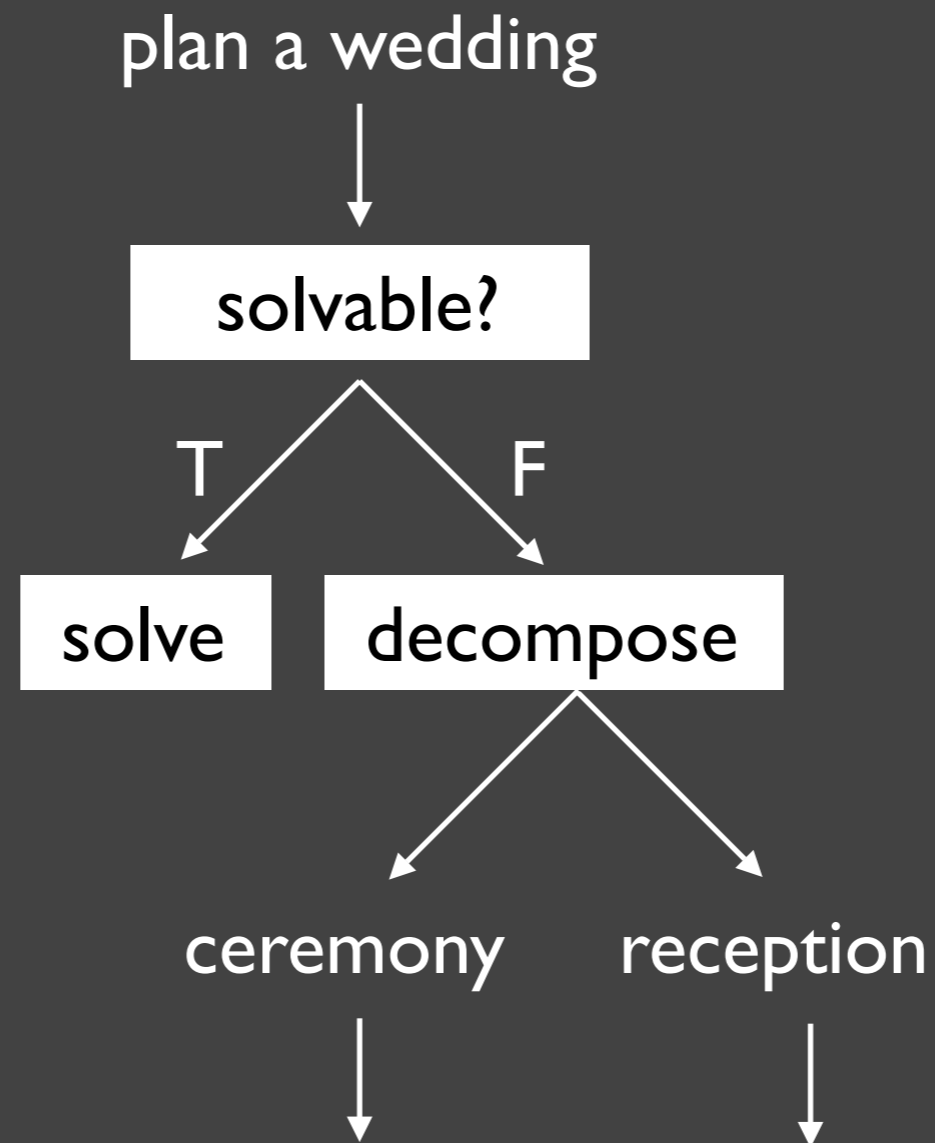
Human Computation Algorithms

human-driven program synthesis



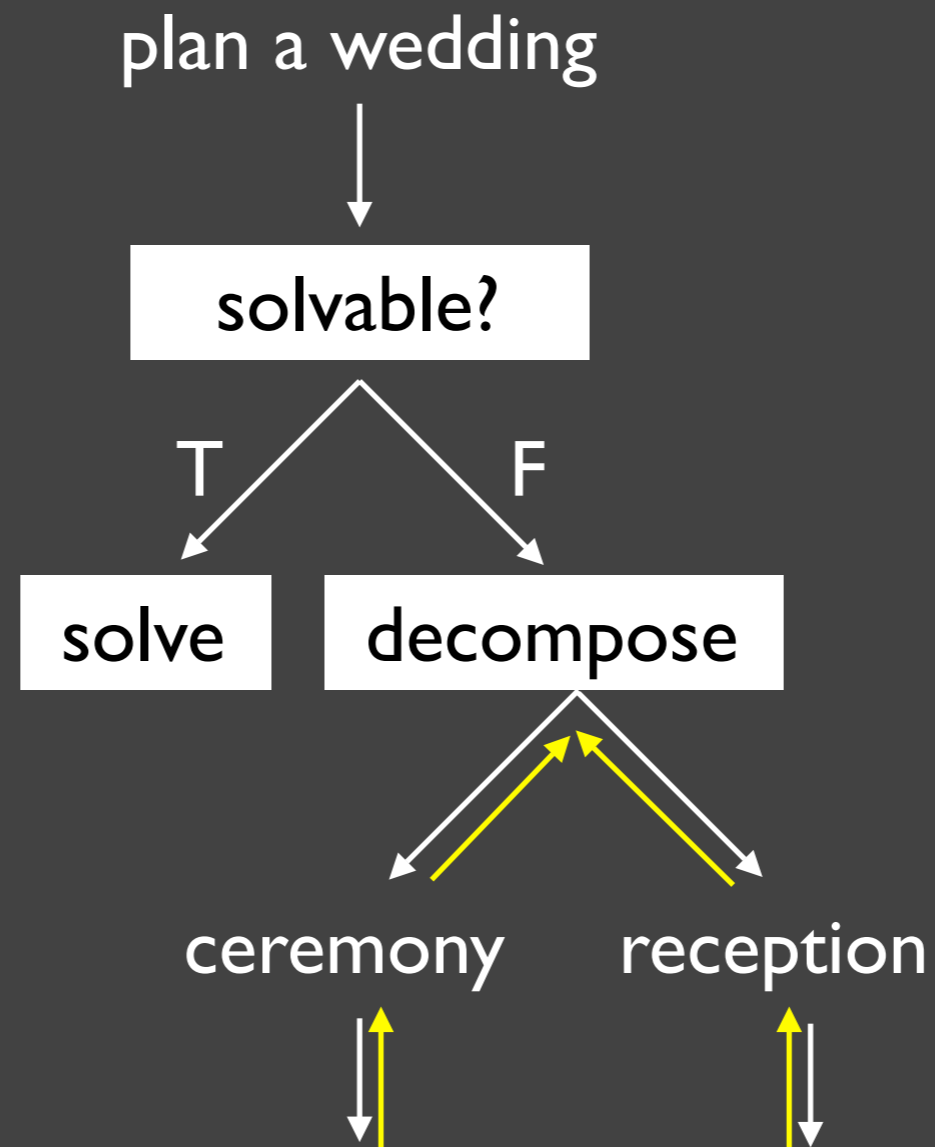
Human Computation Algorithms

human-driven program synthesis



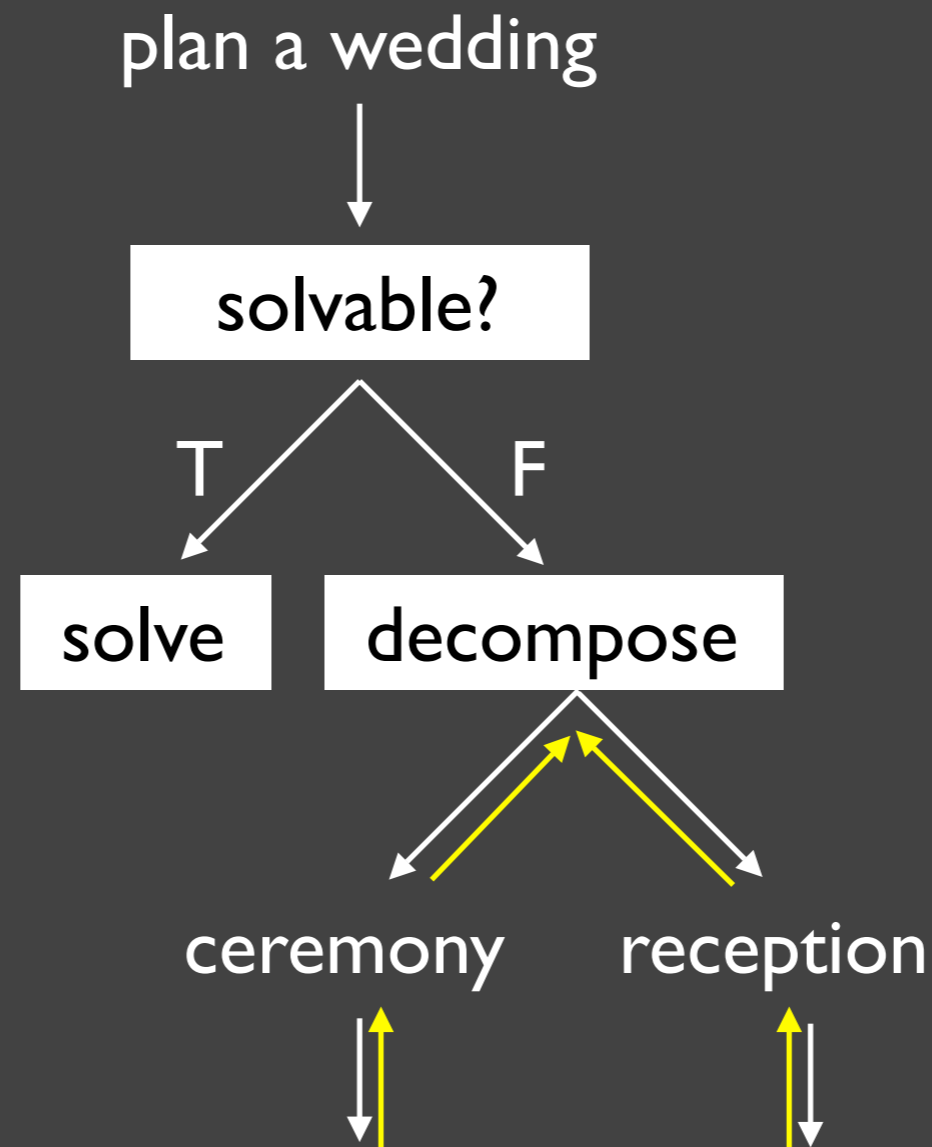
Human Computation Algorithms

human-driven program synthesis



Human Computation Algorithms

human-driven program synthesis

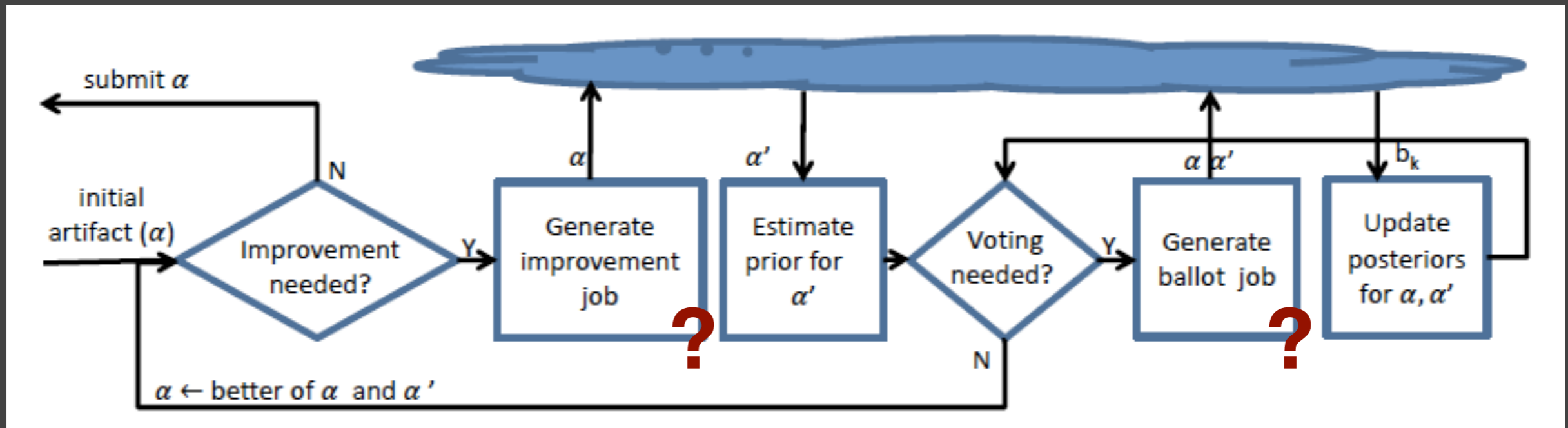


Turkomatic (Kulkarni et al., 2011)
CrowdForge (Kittur et al., 2011)

Human Computation Algorithms

automated design

Another improvement task? How many votes?



TurKontrol (Dai et al., 2010; 2011)

PROPERTIES

Is the algorithm **correct**?

Correctness

Theoretical Analysis

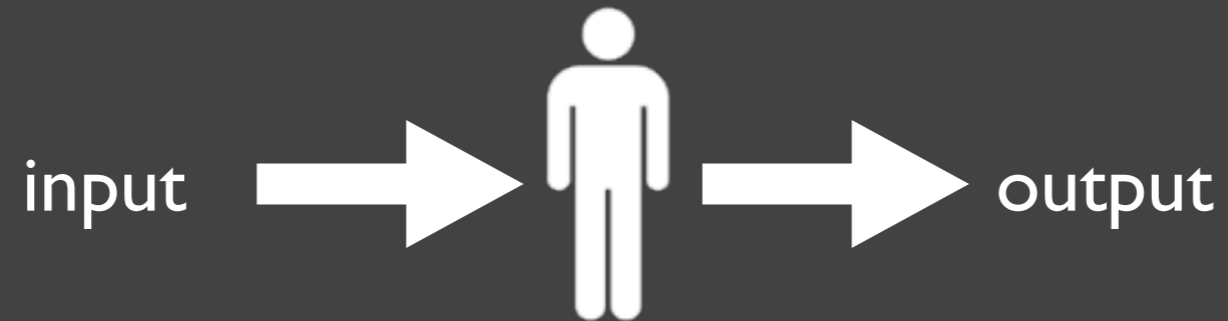
What does it mean for a human computation algorithm to be correct?

What **guarantees** can we give regarding the correctness of a human computation algorithm?



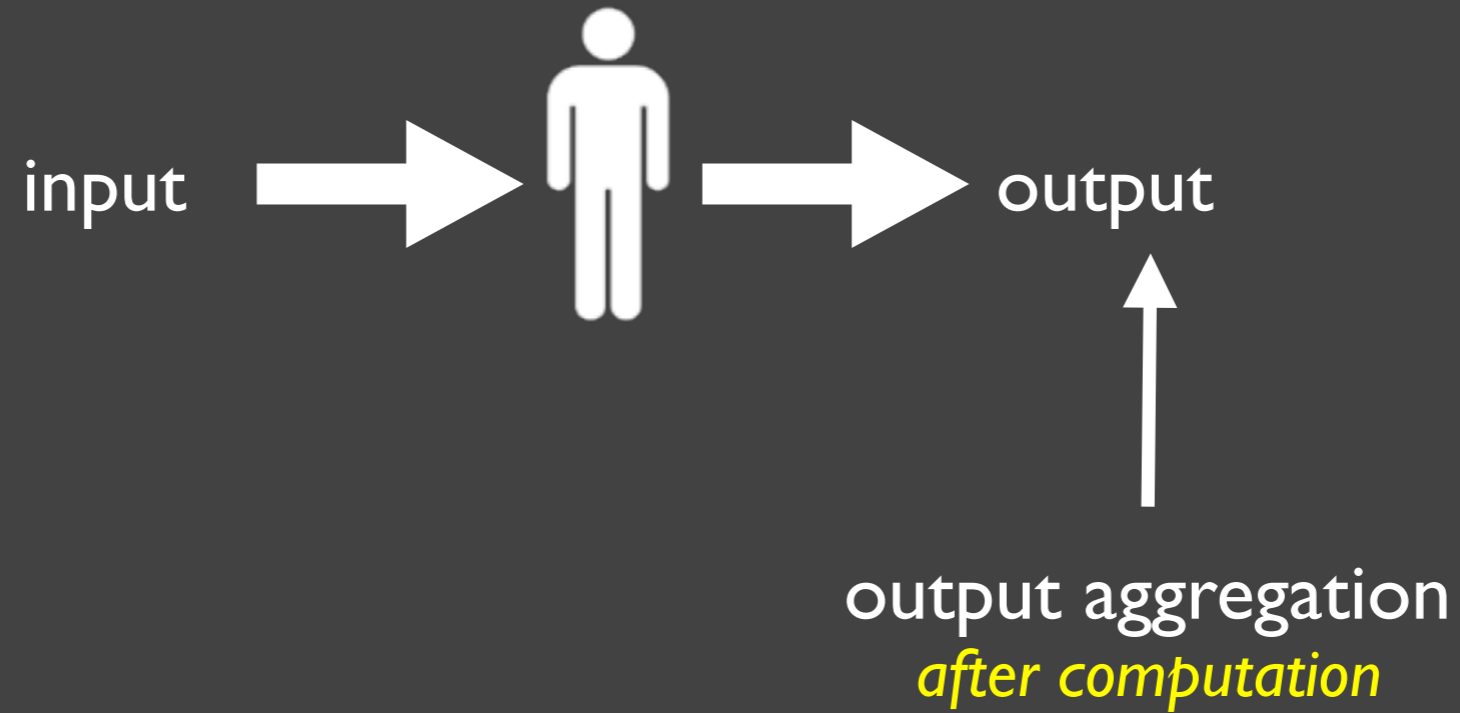
Correctness

Three Points of Intervention



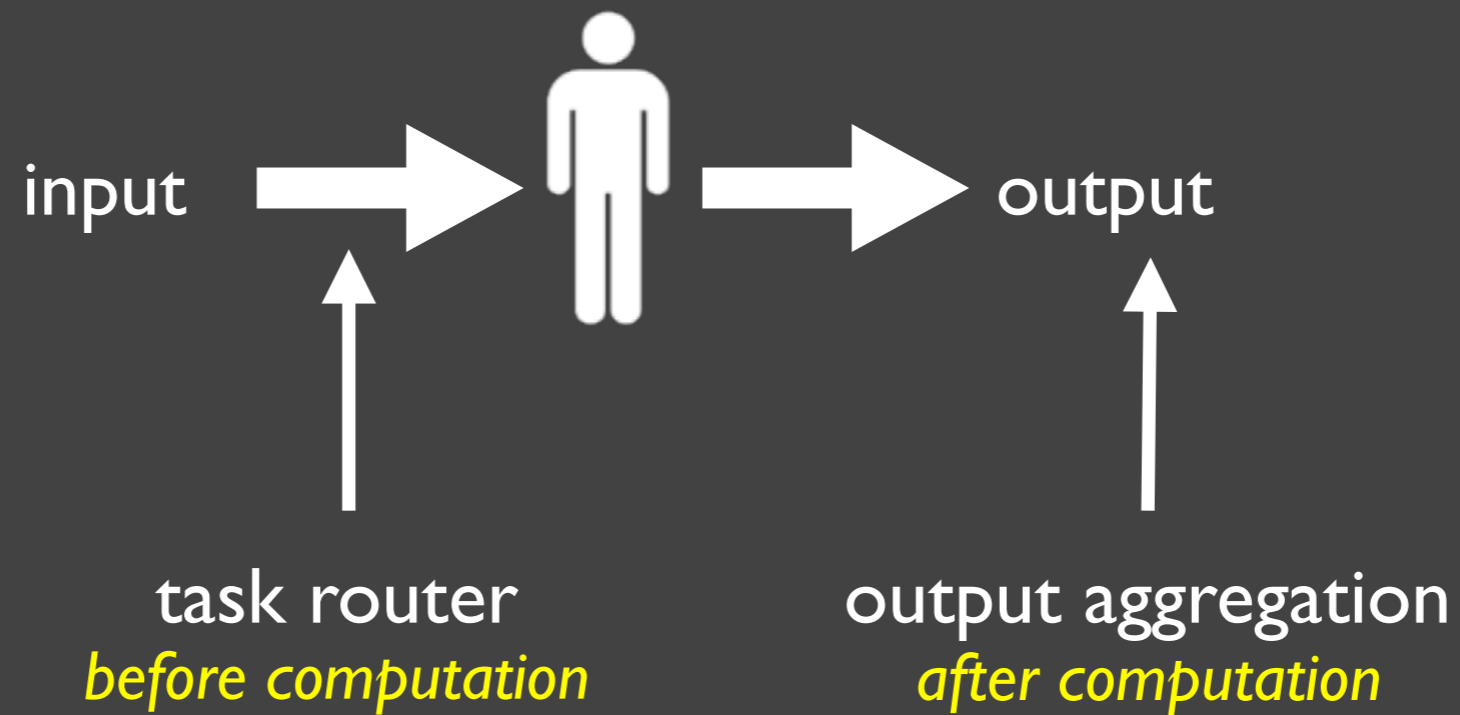
Correctness

Three Points of Intervention



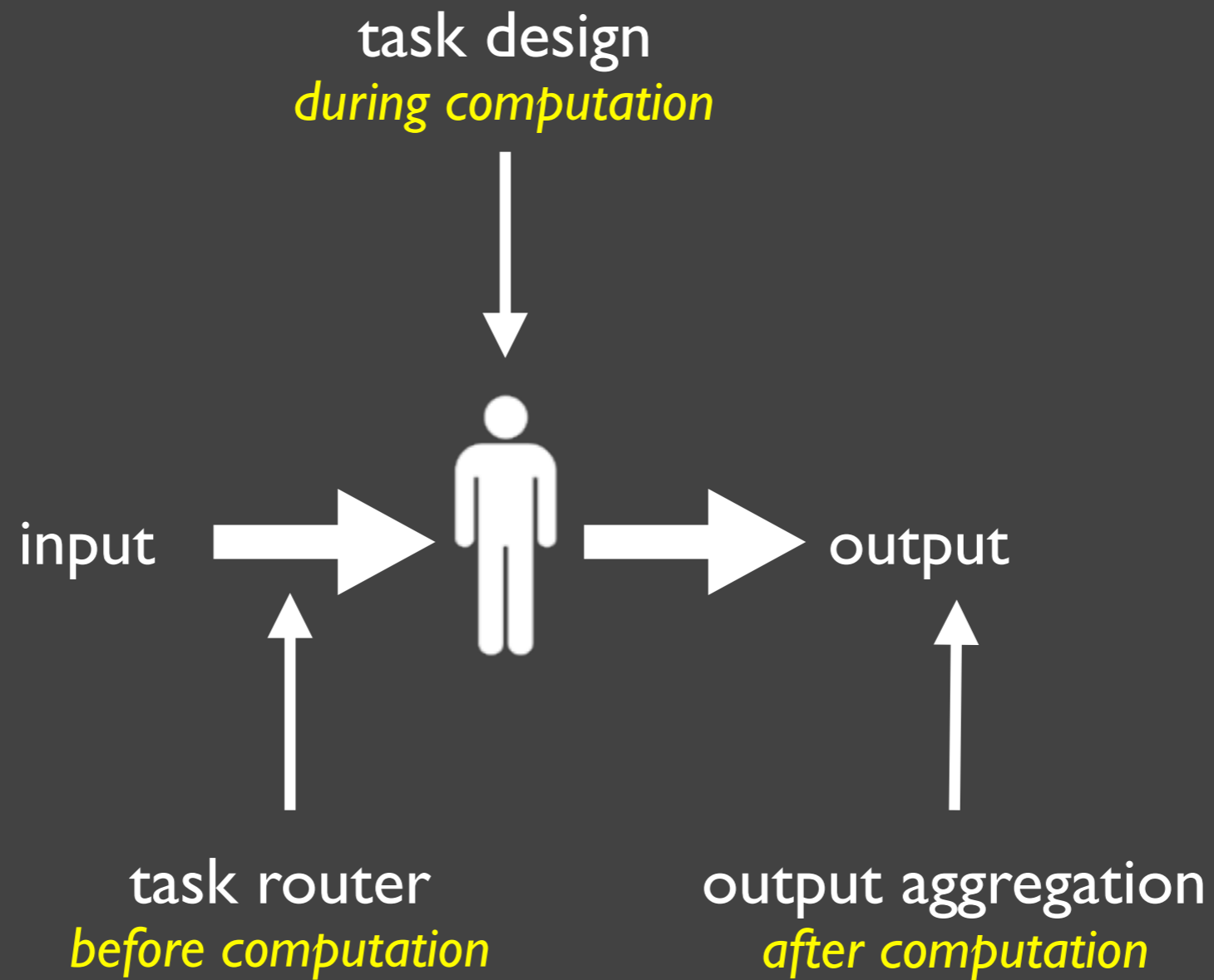
Correctness

Three Points of Intervention



Correctness

Three Points of Intervention



Is the algorithm **efficient**?

Efficiency

Three Measures

Time Complexity

How long does it take?

Query Complexity

How many queries to the human computers?

Cost Effectiveness

How much does it cost?

Efficiency

Time Complexity

Operation Complexity
How does the number
of operations scale?

VS

Clock Time
How much time does
it actually take?

Efficiency

The need for real-time



vizwiz & quikTurKit (Bigham et al., 2010)



Retainer Model (Bernstein et al., 2011)

Efficiency

Query Complexity



Efficiency

Query Complexity

I. Repeated Labeling

For each input object, how many human computers do we query?



Efficiency

Query Complexity

I. Repeated Labeling

For each input object, how many human computers do we query?

(Sheng et al., 2009; Kumar and Lease, 2011)



Efficiency

Query Complexity

Efficiency

Query Complexity

2. Active Learning

Which input should we process? What questions should we ask?

Active Learning

a short introduction

“The learner can **select the data** from which it learns.”
(Settles, 2011)

Active Learning

a short introduction

“The learner can **select the data** from which it learns.”
(Settles, 2011)

a single perfect oracle

label / feature / feature value

Active Learning

a short introduction

“The learner can **select the data** from which it learns.”
(Settles, 2011)

a single ~~perfect~~ oracle

label / feature / feature value

Active Learning

a short introduction

“The learner can **select the data** from which it learns.”
(Settles, 2011)

multiple imperfect oracles
a single ~~pe~~fect oracle

label / feature / feature value

Active Learning

a short introduction

“The learner can **select the data** from which it learns.”
(Settles, 2011)

multiple imperfect oracles
a single ~~pe~~fect oracle

label / ~~feature~~ feature value

Active Learning

a short introduction

“The learner can **select the data** from which it learns.”
(Settles, 2011)

multiple imperfect oracles

a single perfect oracle

richer, different kinds of questions

label / feature ~~value~~ feature value

Active Learning

example # 1



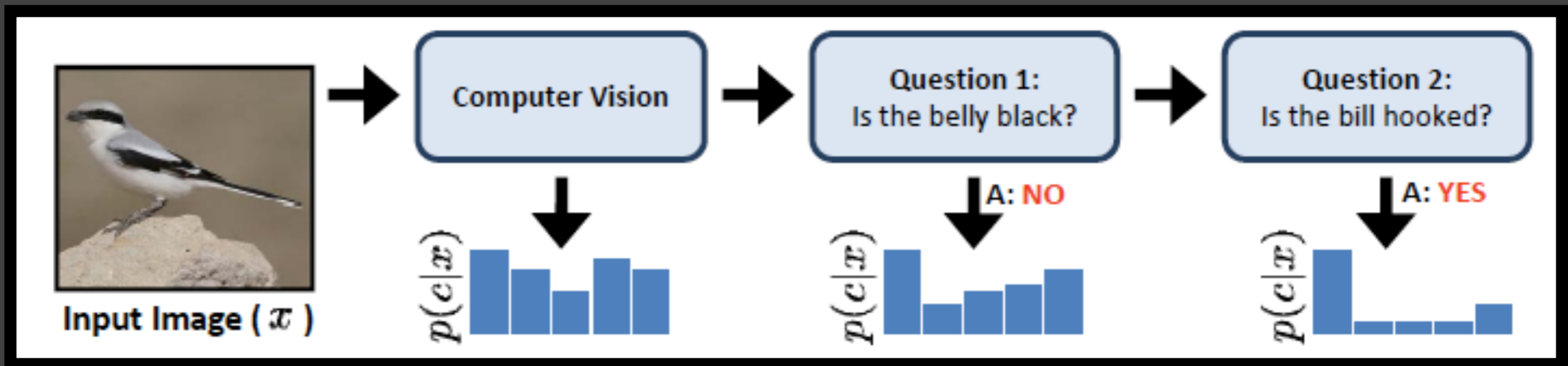
(a)

(b)



(Tamuz et al., 2011)

Active Learning example # 2



(Branson et al., 2010)

Efficiency

Cost Effectiveness



Efficiency

Cost Effectiveness



How do we price each task?

Will the total cost be within budget?

What is the total cost in the worst case?

Can we minimize cost?

What is the cost-benefit tradeoff?



RECAP

TAKE-HOME

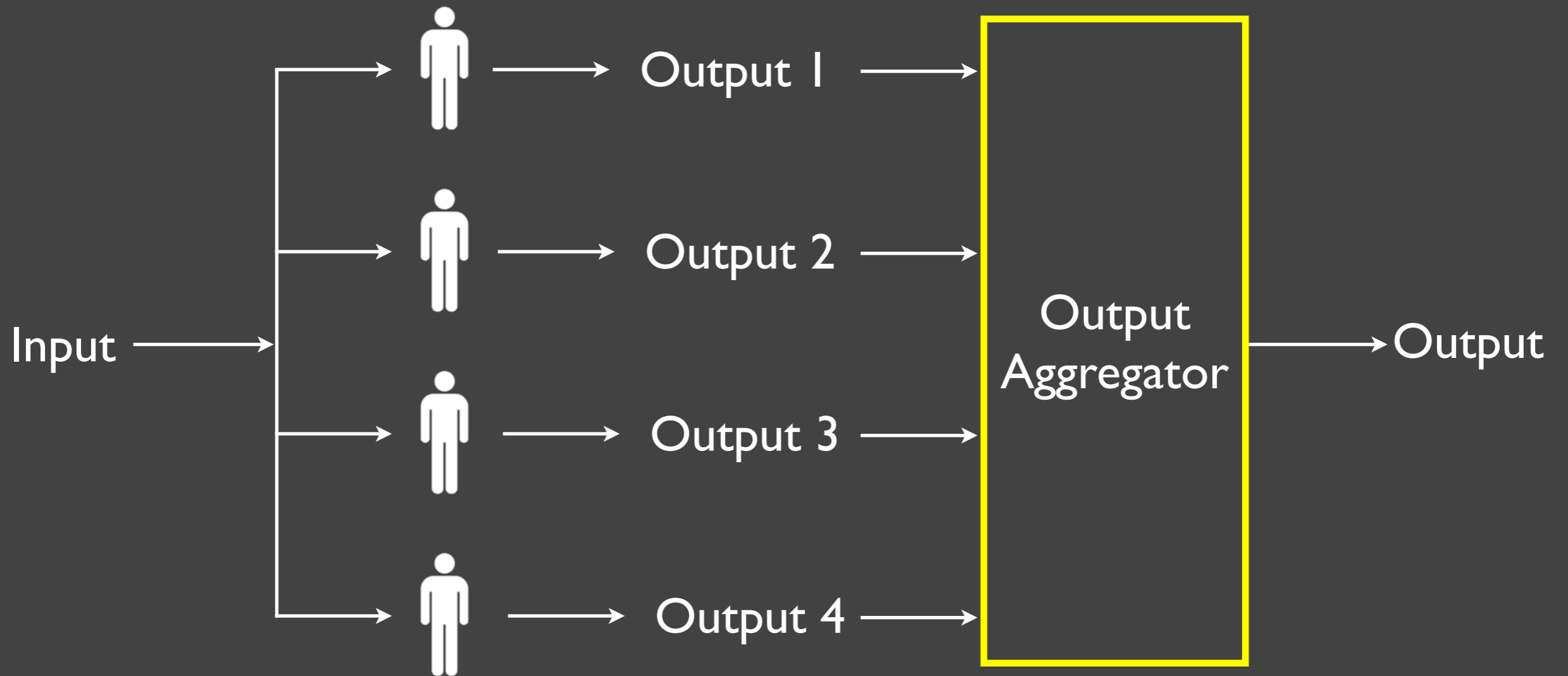
“human computation algorithms \leftrightarrow automated algorithms”

V

OUTPUT AGGREGATION

Motivation • Simple Outputs • Complex Outputs

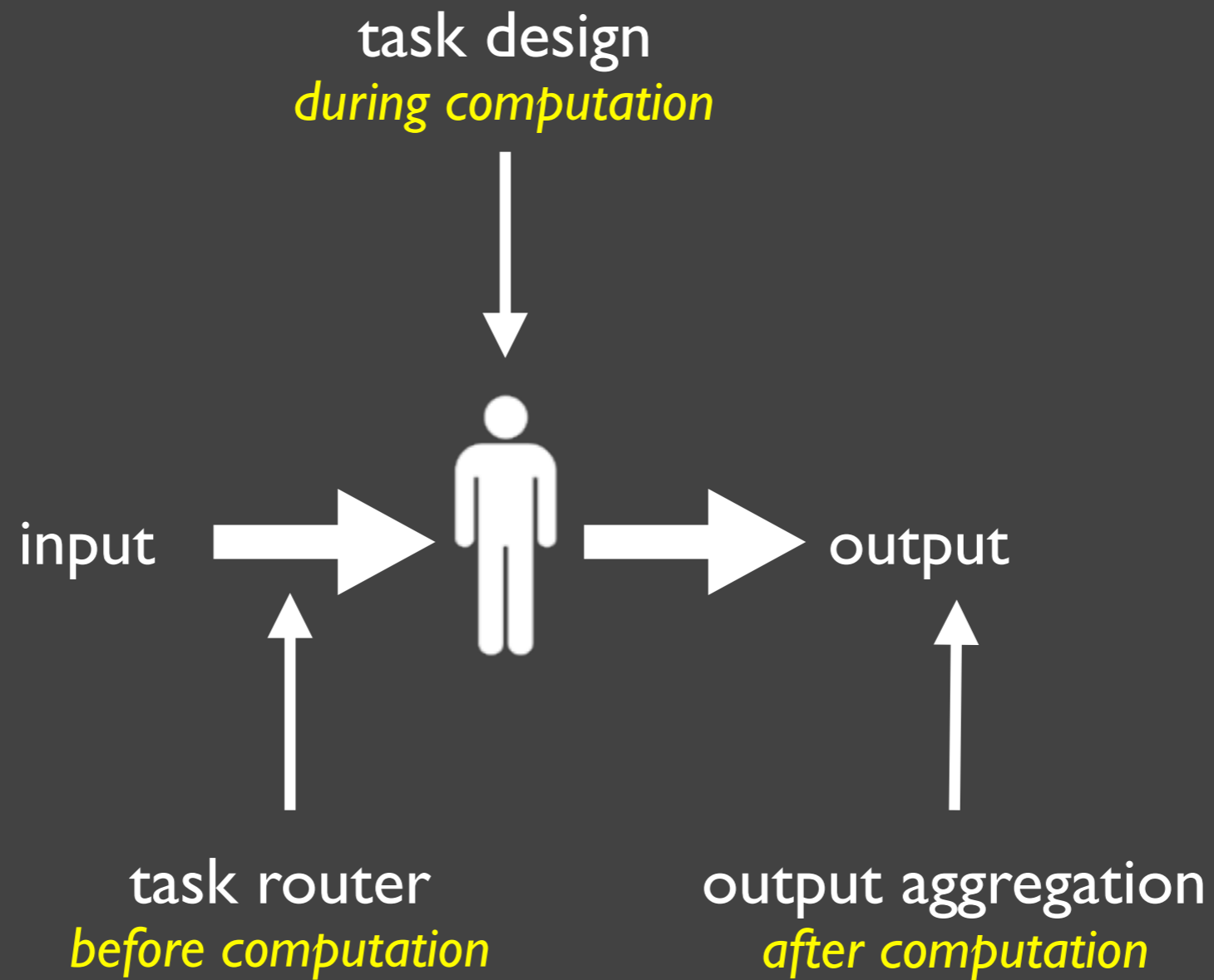
Output Aggregation in a nutshell



Outputs can be aggregated by humans or **automatically**.

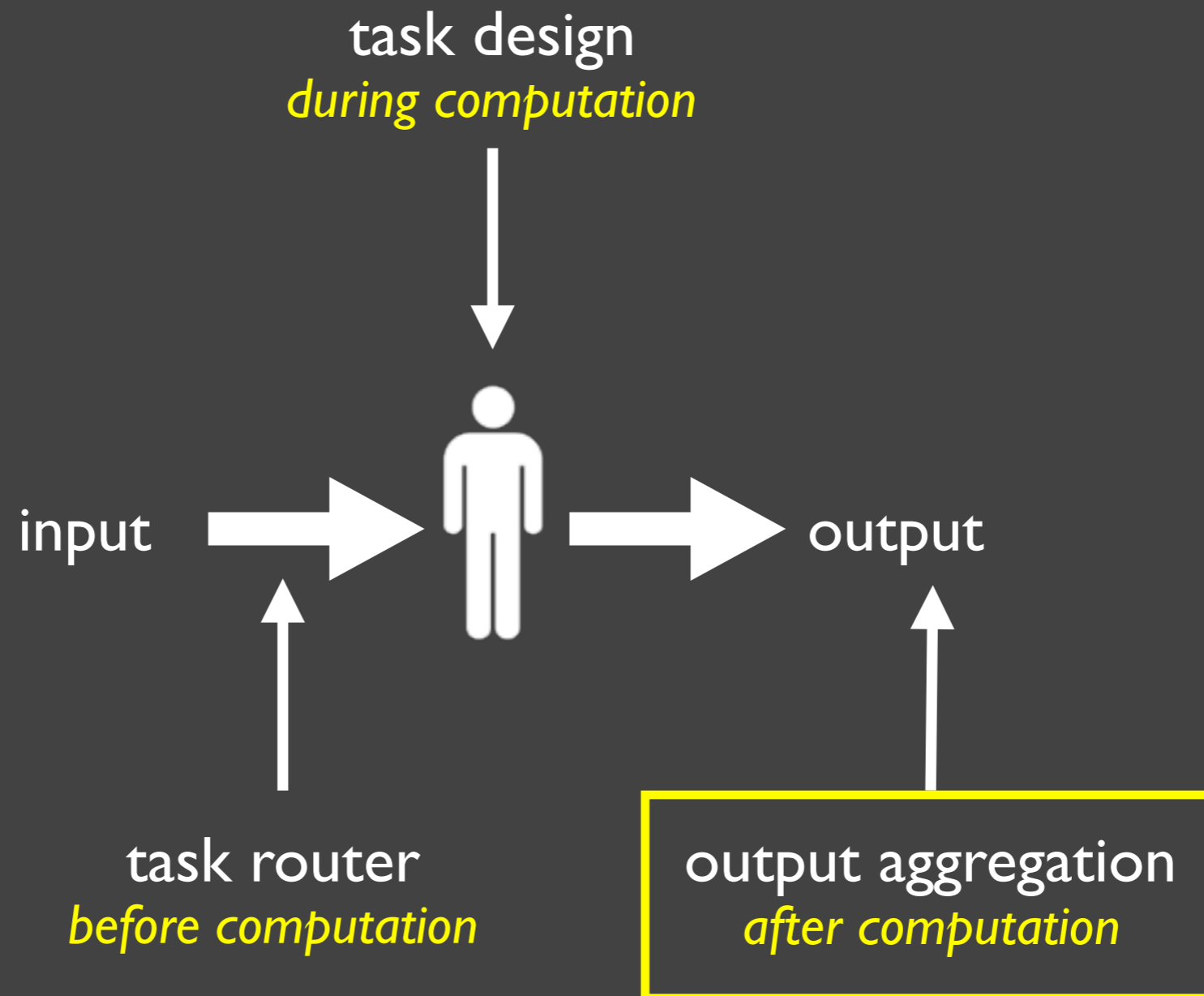
Correctness

Three Points of Intervention



Correctness

Three Points of Intervention



MOTIVATION

Outputs generated by human computers can be **noisy**.

Noise

is not only about inaccuracy

Noise

is not only about inaccuracy

The screenshot shows the 'Tag a Tune' game interface. At the top, the score is 80 and the timer is 1:41. A 'Bonus' progress bar is visible. The main area is split into two columns: 'Describe the tune ...' and 'Listening to the same tune?'. The 'Describe the tune ...' column has a play button and a progress bar. The 'Listening to the same tune?' column has 'same' and 'different' buttons, with 'different' selected and a '1 in a row!' indicator. Below this, there are two columns of descriptions: 'your descriptions' and 'your partner's descriptions'. A central overlay shows 'Correct' with '60 points' and green checkmarks for both 'You' and 'Partner'. The 'Your partner has chosen.' message is at the bottom right.

Score 80

Tag a Tune Hear Here

Timer 1:41

Bonus

Describe the tune ...

Listening to the same tune?

same different 1 in a row!

your descriptions

male vocal

medieval music

quartet

two females

your partner's descriptions

guitar

solo

no vocals

You Correct Partner

60 points

Your partner has chosen.

+ submit → pass

(Law and von Ahn, 2009)

Noise

example from TagATune

CLASSICAL	GRUITAR	FEMALE	RENNAISSANCE	STOMP
GUITAR	PRIMAL	VOCAL	SWING	SKIPPY
PIANO	ACCUSTIC	QUIET	SCI-FI	FOREIGN
VIOLIN	ACTIVE	SITAR	HIPPIE	CHRISTMASSY
ROCK	MEOW	CLASSIC	LULLABY	CLAPPY
SLOW	OHOHOH	SOFT	ANGELIC	CLOUDY
STRINGS	GRADUAL	CELLO	DOWNBEAT	SEASIDE
TECHNO	CLIMATIC	WOMAN	RELAXATION	MAMBO
OPERA	PENSIVE	MALE	GLOOMY	MANDOLIN
DRUMS	HOUSY	SINGING	ROYAL	FOLK
SAME	INSTRUMENTAL	VOCALS	RYTHMIC	NO VIOLINS
FLUTE	CALMISH	SOLO	MUFFLED	MELODY
FAST	FEMALE OPERA	LOUD	RAGTIME	HARMONICA
DIFF	VARIED	CHOIR	TUDOR	ITALIAN
ELECTRONIC	HEALING	VIOLINS	FANTASY	DRAMATIC
AMBIENT	WAVEY	HARP	HISPANIC	BLUEGRASS
BEAT	DRIPPING	BEATS	BEATLES	GENTLE
HARPSICORD	HEBREW	NOT ROCK	SYNCOATED	SPACESHIP DESCENDING
SYNTH	ANIMALS	WIERD	MID-TEMPO	COOKIE MONSTER VOCAL
INDIAN	REEDS	DANCE	RATTLE	VAMPIRES AT A DINNER PARTY

(Law, Settles and Mitchell, 2010)

Noise

example from TagATune

CLASSICAL	GRUITAR	FEMALE	RENNAISSANCE	STOMP
GUITAR	PRIMAL	VOCAL	SWING	SKIPPY
PIANO	ACCUSTIC	QUIET	SCI-FI	FOREIGN
VIOLIN	ACTIVE	SITAR	HIPPIE	CHRISTMASSY
ROCK	MEOW	CLASSIC	LULLABY	CLAPPY
SLOW	OHOHOH	SOFT	ANGELIC	CLOUDY
STRINGS	GRADUAL	CELLO	DOWNBEAT	SEASIDE
TECHNO	CLIMATIC	WOMAN	RELAXATION	MAMBO
OPERA	PENSIVE	MALE	GLOOMY	MANDOLIN
DRUMS	HOUSY	SINGING	ROYAL	FOLK
SAME	INSTRUMENTAL	VOCALS	RYTHMIC	NO VIOLINS
FLUTE	CALMISH	SOLO	MUFFLED	MELODY
FAST	FEMALE OPERA	LOUD	RAGTIME	HARMONICA
DIFF	VARIED	CHOIR	TUDOR	ITALIAN
ELECTRONIC	HEALING	VIOLINS	FANTASY	DRAMATIC
AMBIENT	WAVEY	HARP	HISPANIC	BLUEGRASS
BEAT	DRIPPING	BEATS	BEATLES	GENTLE
HARPSICORD	HEBREW	NOT ROCK	SYNCOATED	SPACESHIP DESCENDING
SYNTH	ANIMALS	WIERD	MID-TEMPO	COOKIE MONSTER VOCAL
INDIAN	REEDS	DANCE	RATTLE	VAMPIRES AT A DINNER PARTY

(Law, Settles and Mitchell, 2010)

Noise

example from TagATune

CLASSICAL	GRUITAR	FEMALE	RENNAISSANCE	STOMP
GUITAR	PRIMAL	VOCAL	SWING	SKIPPY
PIANO	ACCUSTIC	QUIET	SCI-FI	FOREIGN
VIOLIN	ACTIVE	SITAR	HIPPIE	CHRISTMASSY
ROCK	MEOW	CLASSIC	LULLABY	CLAPPY
SLOW	OHOHOH	SOFT	ANGELIC	CLOUDY
STRINGS	GRADUAL	CELLO	DOWNBEAT	SEASIDE
TECHNO	CLIMATIC	WOMAN	RELAXATION	MAMBO
OPERA	PENSIVE	MALE	GLOOMY	MANDOLIN
DRUMS	HOUSY	SINGING	ROYAL	FOLK
SAME	INSTRUMENTAL	VOCALS	RYTHMIC	NO VIOLINS
FLUTE	CALMISH	SOLO	MUFFLED	MELODY
FAST	FEMALE OPERA	LOUD	RAGTIME	HARMONICA
DIFF	VARIED	CHOIR	TUDOR	ITALIAN
ELECTRONIC	HEALING	VIOLINS	FANTASY	DRAMATIC
AMBIENT	WAVEY	HARP	HISPANIC	BLUEGRASS
BEAT	DRIPPING	BEATS	BEATLES	GENTLE
HARPSICORD	HEBREW	NOT ROCK	SYNCOATED	SPACESHIP DESCENDING
SYNTH	ANIMALS	WIERD	MID-TEMPO	COOKIE MONSTER VOCAL
INDIAN	REEDS	DANCE	RATTLE	VAMPIRES AT A DINNER PARTY

(Law, Settles and Mitchell, 2010)

Noise

example from TagATune

CLASSICAL	GRUITAR	FEMALE	RENNAISSANCE	STOMP
GUITAR	PRIMAL	VOCAL	SWING	SKIPPY
PIANO	ACCUSTIC	QUIET	SCI-FI	FOREIGN
VIOLIN	ACTIVE	SITAR	HIPPIE	CHRISTMASSY
ROCK	MEOW	CLASSIC	LULLABY	CLAPPY
SLOW	OHOHOH	SOFT	ANGELIC	CLOUDY
STRINGS	GRADUAL	CELLO	DOWNBEAT	SEASIDE
TECHNO	CLIMATIC	WOMAN	RELAXATION	MAMBO
OPERA	PENSIVE	MALE	GLOOMY	MANDOLIN
DRUMS	HOUSY	SINGING	ROYAL	FOLK
SAME	INSTRUMENTAL	VOCALS	RYTHMIC	NO VIOLINS
FLUTE	CALMISH	SOLO	MUFFLED	MELODY
FAST	FEMALE OPERA	LOUD	RAGTIME	HARMONICA
DIFF	VARIED	CHOIR	TUDOR	ITALIAN
ELECTRONIC	HEALING	VIOLINS	FANTASY	DRAMATIC
AMBIENT	WAVEY	HARP	HISPANIC	BLUEGRASS
BEAT	DRIPPING	BEATS	BEATLES	GENTLE
HARPSICORD	HEBREW	NOT ROCK	SYNCOATED	SPACESHIP DESCENDING
SYNTH	ANIMALS	WIERD	MID-TEMPO	COOKIE MONSTER VOCAL
INDIAN	REEDS	DANCE	RATTLE	VAMPIRES AT A DINNER PARTY

(Law, Settles and Mitchell, 2010)

Noise

example from TagATune

CLASSICAL	GRUITAR	FEMALE	RENNAISSANCE	STOMP
GUITAR	PRIMAL	VOCAL	SWING	SKIPPY
PIANO	ACCUSTIC	QUIET	SCI-FI	FOREIGN
VIOLIN	ACTIVE	SITAR	HIPPIE	CHRISTMASSY
ROCK	MEOW	CLASSIC	LULLABY	CLAPPY
SLOW	OHOHOH	SOFT	ANGELIC	CLOUDY
STRINGS	GRADUAL	CELLO	DOWNBEAT	SEASIDE
TECHNO	CLIMATIC	WOMAN	RELAXATION	MAMBO
OPERA	PENSIVE	MALE	GLOOMY	MANDOLIN
DRUMS	HOUSY	SINGING	ROYAL	FOLK
SAME	INSTRUMENTAL	VOCALS	RYTHMIC	NO VIOLINS
FLUTE	CALMISH	SOLO	MUFFLED	MELODY
FAST	FEMALE OPERA	LOUD	RAGTIME	HARMONICA
DIFF	VARIED	CHOIR	TUDOR	ITALIAN
ELECTRONIC	HEALING	VIOLINS	FANTASY	DRAMATIC
AMBIENT	WAVEY	HARP	HISPANIC	BLUEGRASS
BEAT	DRIPPING	BEATS	BEATLES	GENTLE
HARPSICORD	HEBREW	NOT ROCK	SYNCOATED	SPACESHIP DESCENDING
SYNTH	ANIMALS	WIERD	MID-TEMPO	COOKIE MONSTER VOCAL
INDIAN	REEDS	DANCE	RATTLE	VAMPIRES AT A DINNER PARTY

(Law, Settles and Mitchell, 2010)

Noise

example from TagATune

CLASSICAL	GRUITAR	FEMALE	RENAISSANCE	STOMP
GUITAR	PRIMAL	VOCAL	SWING	SKIPPY
PIANO	ACCUSTIC	QUIET	SCI-FI	FOREIGN
VIOLIN	ACTIVE	SITAR	HIPPIE	CHRISTMASSY
ROCK	MEOW	CLASSIC	LULLABY	CLAPPY
SLOW	OHOHOH	SOFT	ANGELIC	CLOUDY
STRINGS	GRADUAL	CELLO	DOWNBEAT	SEASIDE
TECHNO	CLIMATIC	WOMAN	RELAXATION	MAMBO
OPERA	PENSIVE	MALE	GLOOMY	MANDOLIN
DRUMS	HOUSY	SINGING	ROYAL	FOLK
SAME	INSTRUMENTAL	VOCALS	RYTHMIC	NO VIOLINS
FLUTE	CALMISH	SOLO	MUFFLED	MELODY
FAST	FEMALE OPERA	LOUD	RAGTIME	HARMONICA
DIFF	VARIED	CHOIR	TUDOR	ITALIAN
ELECTRONIC	HEALING	VIOLINS	FANTASY	DRAMATIC
AMBIENT	WAVEY	HARP	HISPANIC	BLUEGRASS
BEAT	DRIPPING	BEATS	BEATLES	GENTLE
HARPSICORD	HEBREW	NOT ROCK	SYNCOATED	SPACESHIP DESCENDING
SYNTH	ANIMALS	WIERD	MID-TEMPO	COOKIE MONSTER VOCAL
INDIAN	REEDS	DANCE	RATTLE	VAMPIRES AT A DINNER PARTY

(Law, Settles and Mitchell, 2010)

Noise

example from TagATune

CLASSICAL	GRUITAR	FEMALE	RENNAISSANCE	STOMP
GUITAR	PRIMAL	VOCAL	SWING	SKIPPY
PIANO	ACCUSTIC	QUIET	SCI-FI	FOREIGN
VIOLIN	ACTIVE	SITAR	HIPPIE	CHRISTMASSY
ROCK	MEOW	CLASSIC	LULLABY	CLAPPY
SLOW	OHOHOH	SOFT	ANGELIC	CLOUDY
STRINGS	GRADUAL	CELLO	DOWNBEAT	SEASIDE
TECHNO	CLIMATIC	WOMAN	RELAXATION	MAMBO
OPERA	PENSIVE	MALE	GLOOMY	MANDOLIN
DRUMS	HOUSY	SINGING	ROYAL	FOLK
SAME	INSTRUMENTAL	VOCALS	RYTHMIC	NO VIOLINS
FLUTE	CALMISH	SOLO	MUFFLED	MELODY
FAST	FEMALE OPERA	LOUD	RAGTIME	HARMONICA
DIFF	VARIED	CHOIR	TUDOR	ITALIAN
ELECTRONIC	HEALING	VIOLINS	FANTASY	DRAMATIC
AMBIENT	WAVEY	HARP	HISPANIC	BLUEGRASS
BEAT	DRIPPING	BEATS	BEATLES	GENTLE
HARPSICORD	HEBREW	NOT ROCK	SYNCOATED	SPACESHIP DESCENDING
SYNTH	ANIMALS	WIERD	MID-TEMPO	COOKIE MONSTER VOCAL
INDIAN	REEDS	DANCE	RATTLE	VAMPIRES AT A DINNER PARTY

(Law, Settles and Mitchell, 2010)

Noise

example from TagATune

CLASSICAL	GRUITAR	FEMALE	RENNAISSANCE	STOMP
GUITAR	PRIMAL	VOCAL	SWING	SKIPPY
PIANO	ACCUSTIC	QUIET	SCI-FI	FOREIGN
VIOLIN	ACTIVE	SITAR	HIPPIE	CHRISTMASSY
ROCK	MEOW	CLASSIC	LULLABY	CLAPPY
SLOW	OHOHOH	SOFT	ANGELIC	CLOUDY
STRINGS	GRADUAL	CELLO	DOWNBEAT	SEASIDE
TECHNO	CLIMATIC	WOMAN	RELAXATION	MAMBO
OPERA	PENSIVE	MALE	GLOOMY	MANDOLIN
DRUMS	HOUSY	SINGING	ROYAL	FOLK
SAME	INSTRUMENTAL	VOCALS	RYTHMIC	NO VIOLINS
FLUTE	CALMISH	SOLO	MUFFLED	MELODY
FAST	FEMALE OPERA	LOUD	RAGTIME	HARMONICA
DIFF	VARIED	CHOIR	TUDOR	ITALIAN
ELECTRONIC	HEALING	VIOLINS	FANTASY	DRAMATIC
AMBIENT	WAVEY	HARP	HISPANIC	BLUEGRASS
BEAT	DRIPPING	BEATS	BEATLES	GENTLE
HARPSICORD	HEBREW	NOT ROCK	SYNCOATED	SPACESHIP DESCENDING
SYNTH	ANIMALS	WIERD	MID-TEMPO	COOKIE MONSTER VOCAL
INDIAN	REEDS	DANCE	RATTLE	VAMPIRES AT A DINNER PARTY

(Law, Settles and Mitchell, 2010)

Noise

example from TagATune

CLASSICAL	GRUITAR	FEMALE	RENNAISSANCE	STOMP
GUITAR	PRIMAL	VOCAL	SWING	SKIPPY
PIANO	ACCUSTIC	QUIET	SCI-FI	FOREIGN
VIOLIN	ACTIVE	SITAR	HIPPIE	CHRISTMASSY
ROCK	MEOW	CLASSIC	LULLABY	CLAPPY
SLOW	OHOHOH	SOFT	ANGELIC	CLOUDY
STRINGS	GRADUAL	CELLO	DOWNBEAT	SEASIDE
TECHNO	CLIMATIC	WOMAN	RELAXATION	MAMBO
OPERA	PENSIVE	MALE	GLOOMY	MANDOLIN
DRUMS	HOUSY	SINGING	ROYAL	FOLK
SAME	INSTRUMENTAL	VOCALS	RYTHMIC	NO VIOLINS
FLUTE	CALMISH	SOLO	MUFFLED	MELODY
FAST	FEMALE OPERA	LOUD	RAGTIME	HARMONICA
DIFF	VARIED	CHOIR	TUDOR	ITALIAN
ELECTRONIC	HEALING	VIOLINS	FANTASY	DRAMATIC
AMBIENT	WAVEY	HARP	HISPANIC	BLUEGRASS
BEAT	DRIPPING	BEATS	BEATLES	GENTLE
HARPSICORD	HEBREW	NOT ROCK	SYNCOATED	SPACESHIP DESCENDING
SYNTH	ANIMALS	WIERD	MID-TEMPO	COOKIE MONSTER VOCAL
INDIAN	REEDS	DANCE	RATTLE	VAMPIRES AT A DINNER PARTY

(Law, Settles and Mitchell, 2010)

The “truth” exists, and
through **redundancy** we can find it.

Truth

objective versus cultural

Objective Truth

Cultural Truth

Truth

objective versus cultural

Objective Truth

a definitive answer exists beyond human judgments, but hard to reach.

e.g., cancer or not
number of volcanos on Venus
location or time of a photo

Cultural Truth

Truth

objective versus cultural

Objective Truth

a definitive answer exists beyond human judgments, but hard to reach.

e.g., cancer or not
number of volcanos on Venus
location or time of a photo

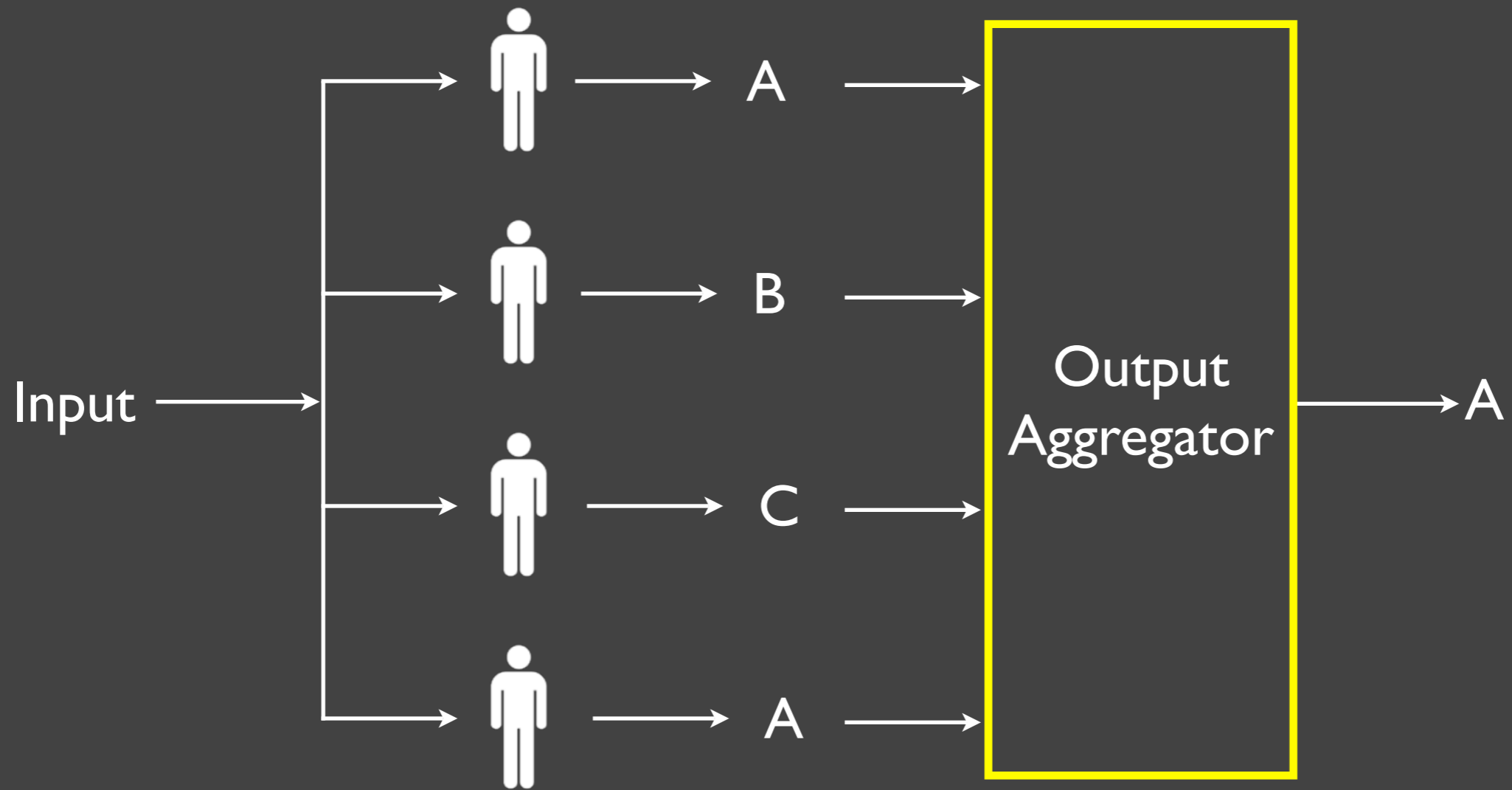
Cultural Truth

shared beliefs of a group of people, often involving perceptual judgments.

e.g., is this music calm?
is this image pornographic?
is this disease contagious?

SIMPLE OUTPUTS

Output Aggregation classification



Statistical Measures of Agreement

(Artstein and Poesio, 2008)

Statistical Measures of Agreement

(Artstein and Poesio, 2008)

The simplest way to aggregate is majority vote.

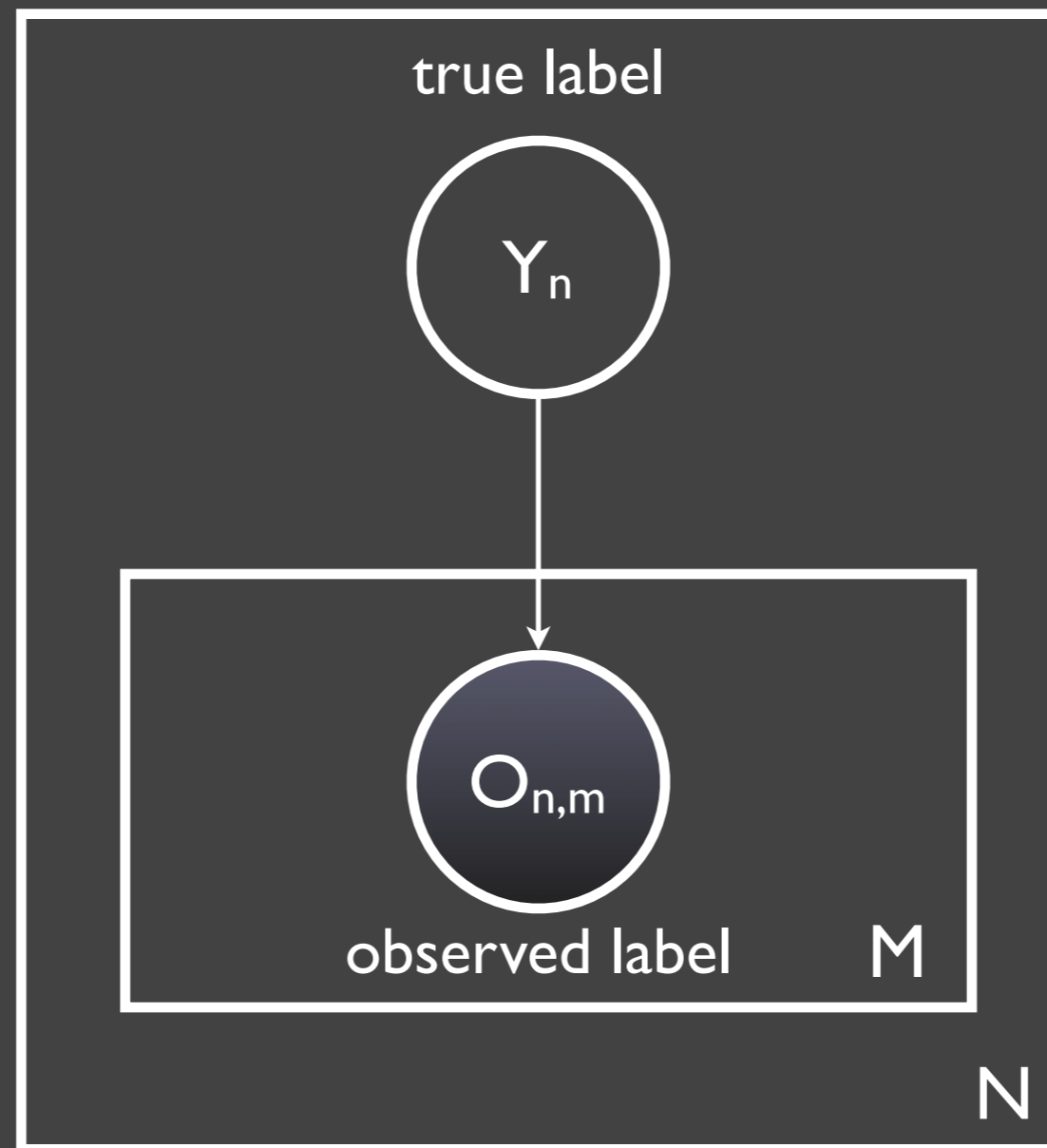
Statistical Measures of Agreement

(Artstein and Poesio, 2008)

The simplest way to aggregate is majority vote.

But how much agreement is really there?

Majority Vote as a graphical model



N classification questions, M workers

Hidden Factors

that influence the annotation process

Hidden Factors that influence the annotation process

Task Characteristics

Quality (e.g., blurry pictures)

Difficulty (e.g., transcription of non-native speech)

Hidden Factors that influence the annotation process

Task Characteristics

Quality (e.g., blurry pictures)

Difficulty (e.g., transcription of non-native speech)

Worker Characteristics

Expertise (e.g., bird identification)

Bias (e.g., mother vs college students)

Physical Conditions (e.g., fatigue)

Latent Class Model for classification

Latent Class Model for classification

Dawid and Skeen, 1979

Uebersax et al., 1993

Carpenter, 2008

Whitehill et al., 2009

Ipeirotis et al., 2010

Raykar et al., 2010

Welinder and Perona, 2010

Ipeirotis et al., 2010

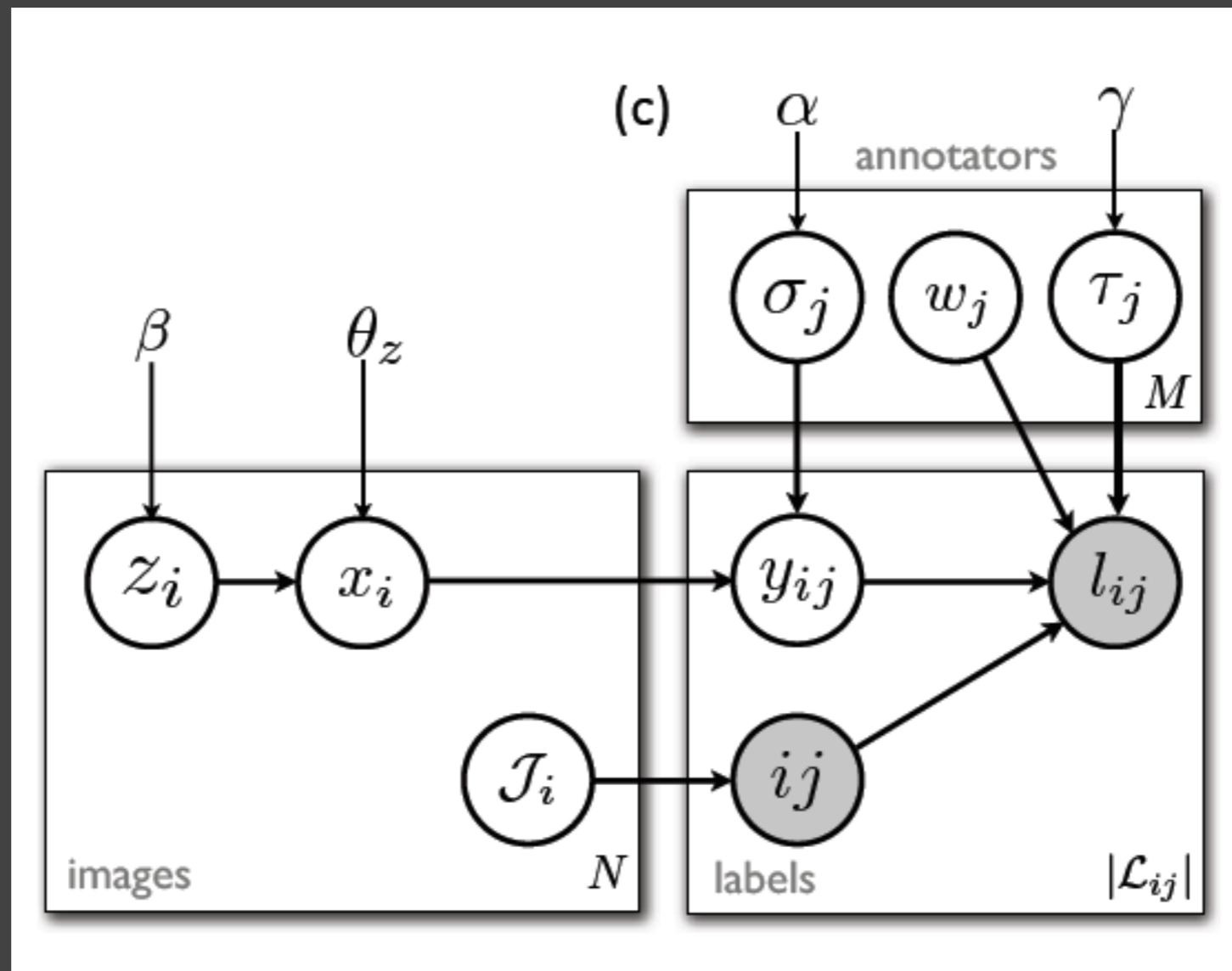
Latent Class Model an example



Grebe?

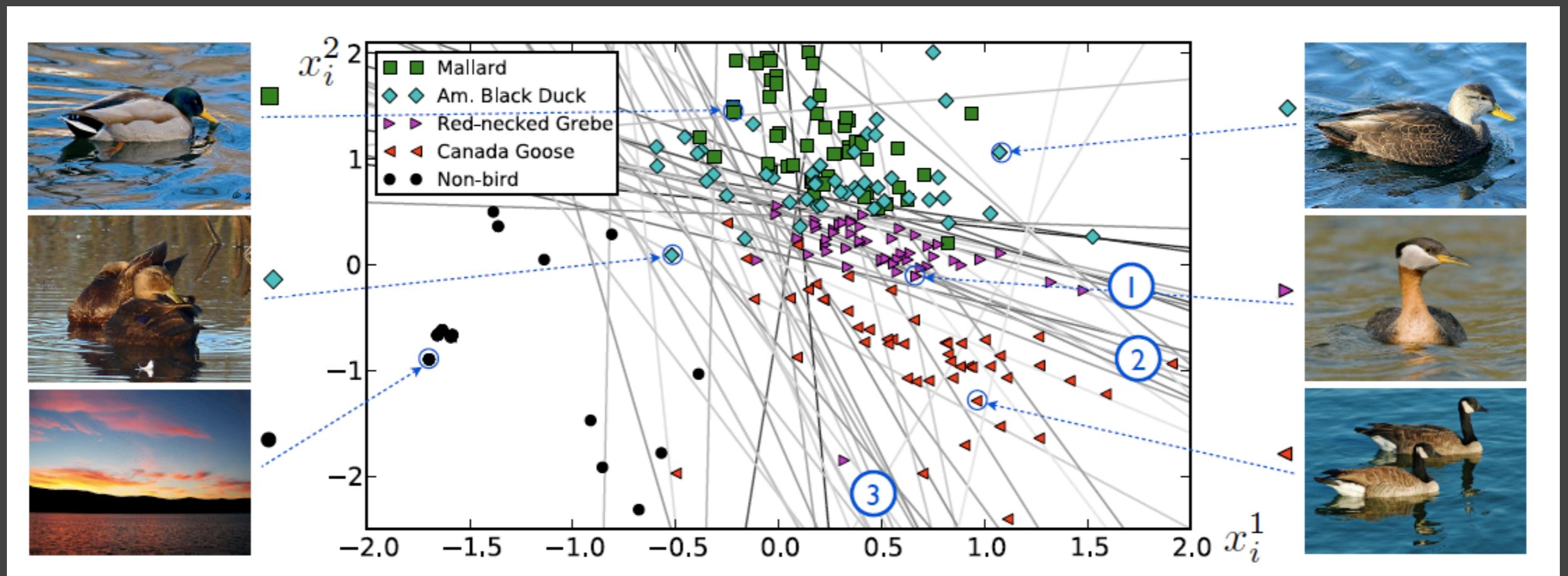
(Welinder et al., 2010)

Latent Class Model an example



(Welinder et al., 2010)

Latent Class Model an example



(Welinder et al., 2010)

Cultural Consensus Theory in Anthropology

	Respondent																							
	10	17	9	23	22	6	18	2	3	5	8	21	1	14	7	13	16	4	20	12	19	15	24	
Allergies	1	1	1	0	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1	0	1	1	1	20**
Kidney pain	1	1	1	1	1	1	1	1	1	0	1	0	1	1	1	1	1	1	0	1	1	1	0	19**
Gastritis	1	1	1	0	1	1	1	1	0	1	1	1	0	1	1	1	1	0	0	0	1	0	1	16*
Amoebas	1	0	1	1	1	1	1	1	0	1	1	1	0	0	1	1	0	0	0	0	1	1	1	15
Appendicitis	1	1	1	1	0	1	0	1	1	1	1	0	1	0	1	1	0	1	1	0	1	0	0	15
Hepatitis	1	1	1	1	1	1	1	1	0	1	1	1	0	1	0	0	0	0	0	1	1	0	14	
Mumps	0	0	0	0	1	0	1	0	1	0	1	0	1	1	1	1	1	1	1	1	1	0	0	13
Rubella	1	1	0	1	1	0	1	0	1	0	0	0	1	1	0	0	1	1	1	1	1	0	0	13
Measles	1	1	0	1	1	0	1	0	1	1	0	0	1	1	0	0	0	1	1	1	1	0	0	13
Smallpox	1	1	1	1	1	0	1	0	1	1	0	0	1	1	0	0	1	0	0	1	1	0	0	13
Cancer	1	1	1	0	0	1	0	1	0	1	1	1	0	0	1	0	0	0	1	0	0	1	1	12
Diabetes	1	1	1	1	0	1	1	1	0	1	1	0	0	0	0	0	0	0	1	0	1	1	0	12
Intestinal influenza	1	0	1	1	1	1	1	1	0	1	1	0	0	0	1	1	0	0	0	0	0	1	0	12
Tetanus	1	1	1	1	0	1	0	1	1	0	1	1	0	0	1	0	0	0	1	0	0	0	1	12
Chicken pox	1	1	0	1	1	0	1	0	1	0	0	0	1	1	0	0	1	1	0	1	1	0	0	12
Tonsillitis	0	0	1	1	0	0	1	1	1	0	1	1	0	1	0	1	0	1	0	0	1	0	0	11
Folio	0	0	1	0	1	1	0	1	1	1	1	1	0	0	1	0	0	0	1	0	0	0	1	11
Diarrhea	0	0	1	1	0	1	1	1	0	0	1	0	0	0	1	1	0	1	0	0	0	1	0	10
Typhoid fever	0	1	0	1	1	0	0	1	1	1	1	0	0	1	0	0	0	1	1	0	0	0	0	10
Diphtheria	0	0	1	0	0	0	0	1	0	0	1	0	0	0	0	0	0	1	1	0	1	0	1	7*
Arthritis	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	1	0	0	0	1	1	1	6**
Whooping cough	1	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	1	1	0	6**
Tuberculosis	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	1	0	1	1	1	6**
Malaria	0	0	0	0	0	0	0	1	0	1	0	0	1	0	0	1	0	0	0	0	1	0	0	5**
Colic	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	1	0	0	0	1	0	4**
Rheumatism	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	3**
Flu	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1**
Errors	2	3	7	6	4	7	4	10	6	6	9	8	4	6	8	8	6	9	9	6	7	12	12	

from (Romney, 1999)

(Romney et al., 1986; Karabatsos and Batchelder, 2003; Weller, 2007)

Other Challenges

What if we cannot assume repeated labeling?

Other Challenges

What if we cannot assume repeated labeling?

“Learn a hypothesis to simulate the aggregate output, and prune away workers that don’t agree”

(Dekel and Shamir, 2009)

COMPLEX OUTPUTS

Complex Outputs and challenges

ranking & clustering

structured outputs

beliefs

Challenge #1:

deciding how to **decompose** the problem

Ranking Aggregation

individual rankings \Rightarrow full ranking

(Cohen et al., 1999; Dwork et al., 2010; Ailon et al., 2005; Fagin et al., 2006)



Ranking Aggregation

individual rankings \Rightarrow full ranking



paired comparison

(Cohen et al., 1999; Dwork et al., 2010; Ailon et al., 2005; Fagin et al., 2006)



Ranking Aggregation

individual rankings \Rightarrow full ranking



paired comparison



rating (scale 1-4)

(Cohen et al., 1999; Dwork et al., 2010; Ailon et al., 2005; Fagin et al., 2006)



Ranking Aggregation

individual rankings \Rightarrow full ranking



paired comparison



rating (scale 1-4)

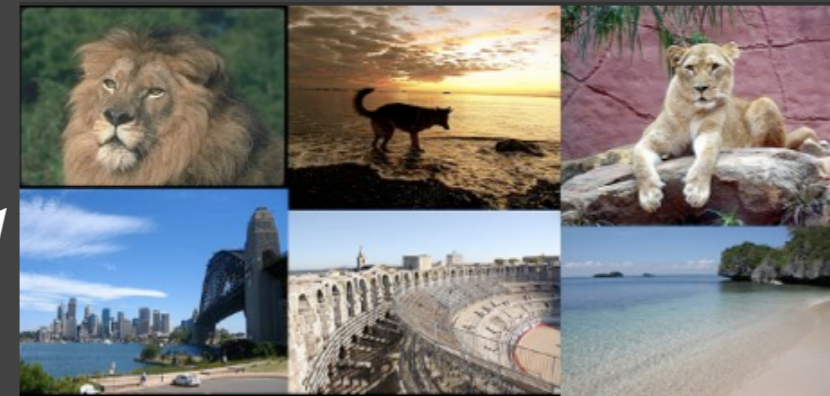
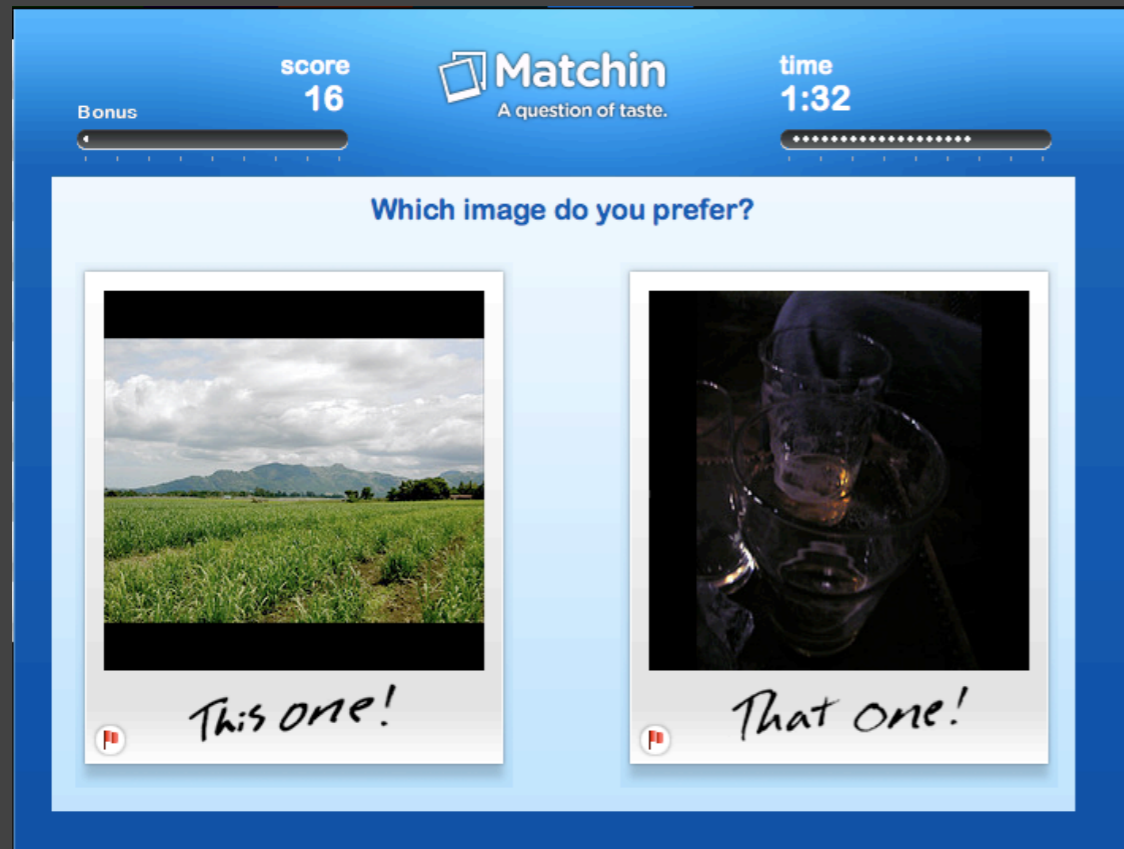


ordering

(Cohen et al., 1999; Dwork et al., 2010; Ailon et al., 2005; Fagin et al., 2006)



Ranking Aggregation an example



(Hacker et al., 2009)

Consensus Clustering

individual clusterings \Rightarrow single clustering

(Topchy et al., 2005; Strehl and Ghosh 2003; Hu and Sung, 2006)



Consensus Clustering

individual clusterings \Rightarrow single clustering



complete clustering

(Topchy et al., 2005; Strehl and Ghosh 2003; Hu and Sung, 2006)



Consensus Clustering

individual clusterings \Rightarrow single clustering



complete clustering



link / cannot-link constraints

(Topchy et al., 2005; Strehl and Ghosh 2003; Hu and Sung, 2006)



Consensus Clustering

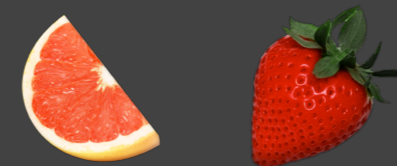
individual clusterings \Rightarrow single clustering



complete clustering



link / cannot-link constraints



how similar are these? (scale 1-5)

(Topchy et al., 2005; Strehl and Ghosh 2003; Hu and Sung, 2006)



Consensus Clustering

an example

(Parent and Eskenazi, 2010)

Consensus Clustering

an example

Do the following definitions of the word aid have the same or different meaning?

- a piece of equipment that helps you to do something.
- something such as a machine or tool that helps someone do something.

LOCAL VIEW

(Parent and Eskenazi, 2010)

Consensus Clustering

an example

Do the following definitions of the word aid have the same or different meaning?

- a piece of equipment that helps you to do something.
- something such as a machine or tool that helps someone do something.

LOCAL VIEW

You have to group the definitions for the word 'code'. There are 2 general meanings.

- to mark a group of things with different colors so that you can tell the difference between them.
- to put a message in code so that it is secret.
- to put a set of numbers, letters, or signs on something to show that it is or give information about it,
- to represent a message in code so that it can only be understood by the person who is meant to receive it.

GLOBAL VIEW

(Parent and Eskenazi, 2010)

Challenge #2:

the **correspondence** problem

Structured Outputs

transcription, translation and description

least difficult



most difficult



Structured Outputs

transcription, translation and description

least difficult

Transcription

ROVER method (Fiscus, 1997)

Longest Common Subsequences, Lattice (Evanini et al., 2010)

most difficult



Structured Outputs

transcription, translation and description

least difficult

Transcription

ROVER method (Fiscus, 1997)

Longest Common Subsequences, Lattice (Evanini et al., 2010)

Translation

BLEU (Papineni et al., 2002);

Consensus Translation (Bangalore et al., 2001; Frederking and Nirenburg, 1994, Matusov et al., 2006, Rosti et al., 2007)

most difficult



Structured Outputs

transcription, translation and description

least difficult

Transcription

ROVER method (Fiscus, 1997)

Longest Common Subsequences, Lattice (Evanini et al., 2010)

Translation

BLEU (Papineni et al., 2002);

Consensus Translation (Bangalore et al., 2001; Frederking and Nirenburg, 1994, Matusov et al., 2006, Rosti et al., 2007)

Description

Information Fushion (Barzilay, 2003; Barzilay et al., 1999)


most difficult



Challenge #3:

aggregating **difficult to articulate** outputs


Belief Aggregation with prediction markets



The Google Lunar X Prize to be won on/before 31 Dec 2012

Last prediction was: **\$4.60 / share**
Today's Change: -

46.0%
CHANCE



Event: [Google Lunar X Prize \(Rule 1.8 Applies\)](#)

Think this event will occur?

Buy Shares

Current best (lowest) price to buy shares is **\$5.00 / share**. There are approx. **1 share** available at this price.

Think this event won't occur?

Sell Shares

Current best (highest) price to sell shares is **\$0.30 / share**. There are approx. **25 shares** available at this price.



RECAP

TAKE-HOME

“classification and **beyond**”

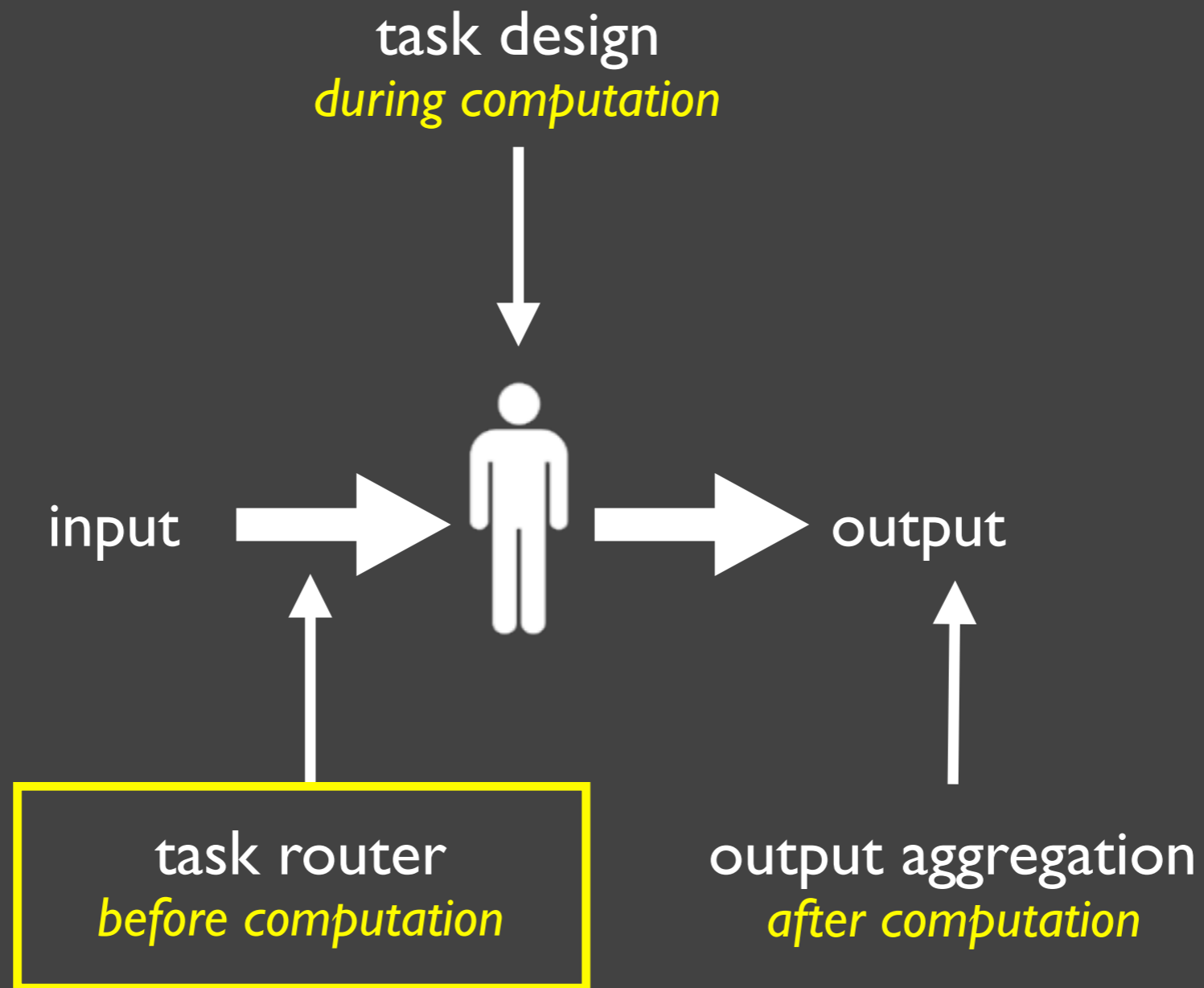
VI

TASK ROUTING

Motivation • Push Methods • Pull Methods

Correctness

Three Points of Intervention



MOTIVATION

The most popular task routing method is WHTBT.

The most popular task routing method is WHTBT.

(which stands for “**Whoever Happens To Be There**”).

All human computers are **not created equal.**

Push versus Pull methods of task routing

Push versus Pull methods of task routing

Push

Workers are passive receivers of tasks.

The system takes complete control over who is assigned which task.

Push versus Pull methods of task routing

Push

Workers are passive receivers of tasks.

The system takes complete control over who is assigned which task.

Pull

Workers are active seekers of tasks.

The system supports a set of interfaces that enable workers to look for tasks to assign themselves.

PUSH METHODS

Push Methods

system → workers

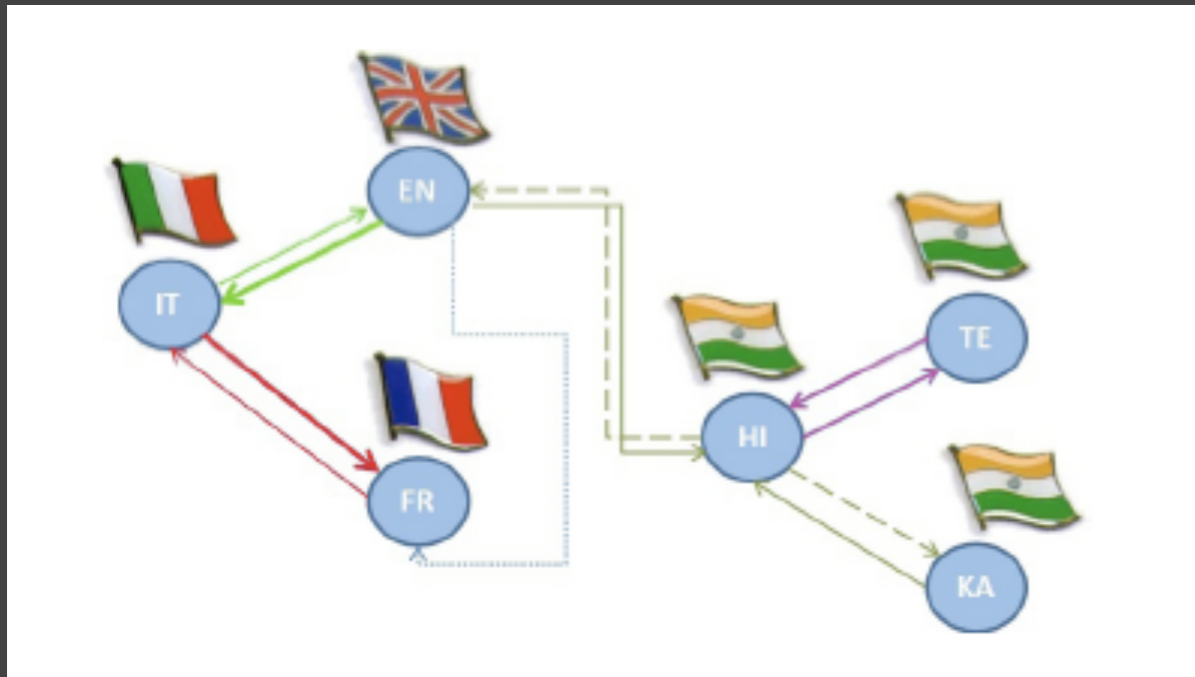
The system takes complete control over who is assigned which task.

Allocation

complete knowledge of utility and cost

Allocation

complete knowledge of utility and cost



(Shahaf and Horvitz, 2010)

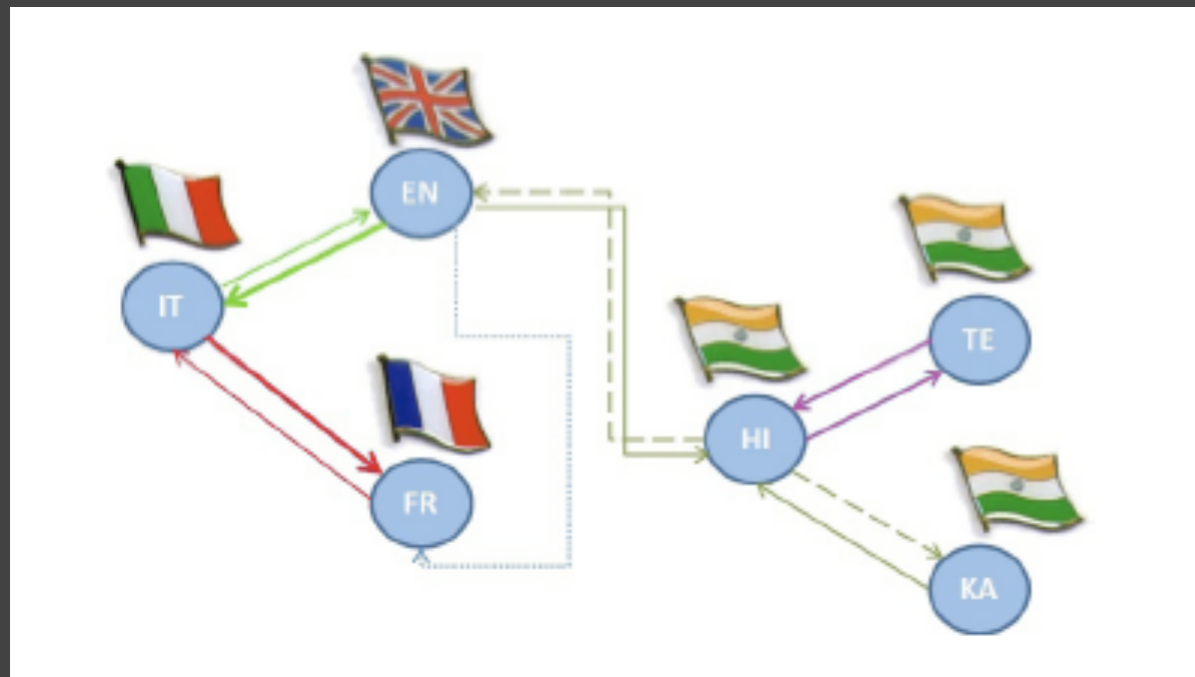
Workers have known competencies.

Tasks have known demands.

weighted exact set-cover problem

Allocation

complete knowledge of utility and cost



(Shahaf and Horvitz, 2010)
Workers have known competencies.
Tasks have known demands.

weighted exact set-cover problem

a.) User Interface on Mobile Phone and Feedback Page on the Web

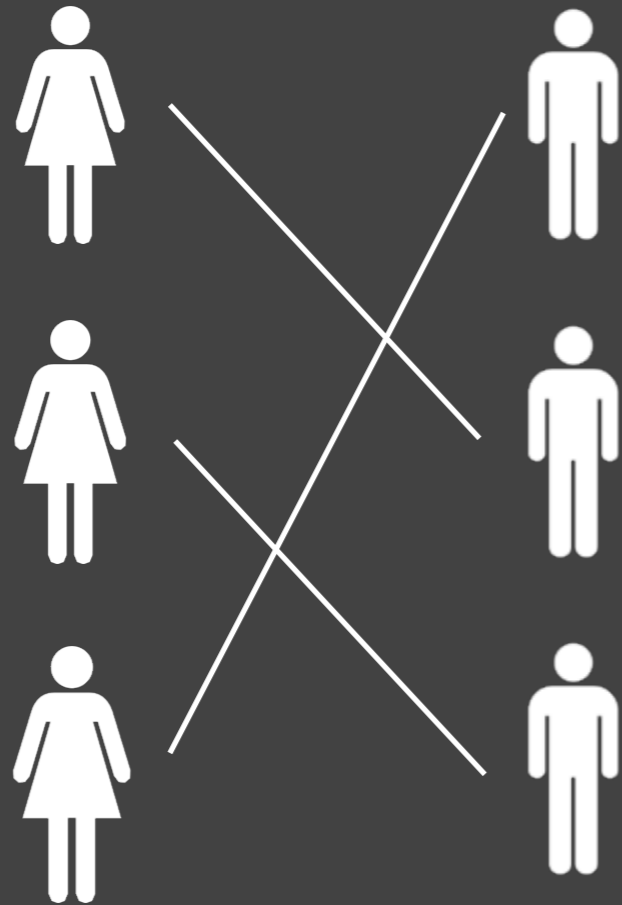
b.) Examples of Images from Sustainability Campaigns

(Reddy et al., 2010)
Participants have known cost and utility
(based on what they can cover).

budgeted maximum coverage problem

Matching

complete or partial preferences



man to woman (Gale and Shapley, 1962)
medical residents to hospitals (Roth, 1984)
students to schools (Teo, 2001)
sailors to ships (Liebowitz, 2000)

incomplete information (Gusfield and Irving, 1989; Liebowitz, 2000)



Inference

incomplete knowledge of utility and cost



Inference

incomplete knowledge of utility and cost

Decision-Theoretic Model

e.g., Donmez et al., 2008

Discovery and Assignment Phase



Inference

incomplete knowledge of utility and cost

Decision-Theoretic Model

e.g., Donmez et al., 2008

Discovery and Assignment Phase

Online Learning

e.g., Donmez et al., 2009

Exploration-Exploitation Tradeoff



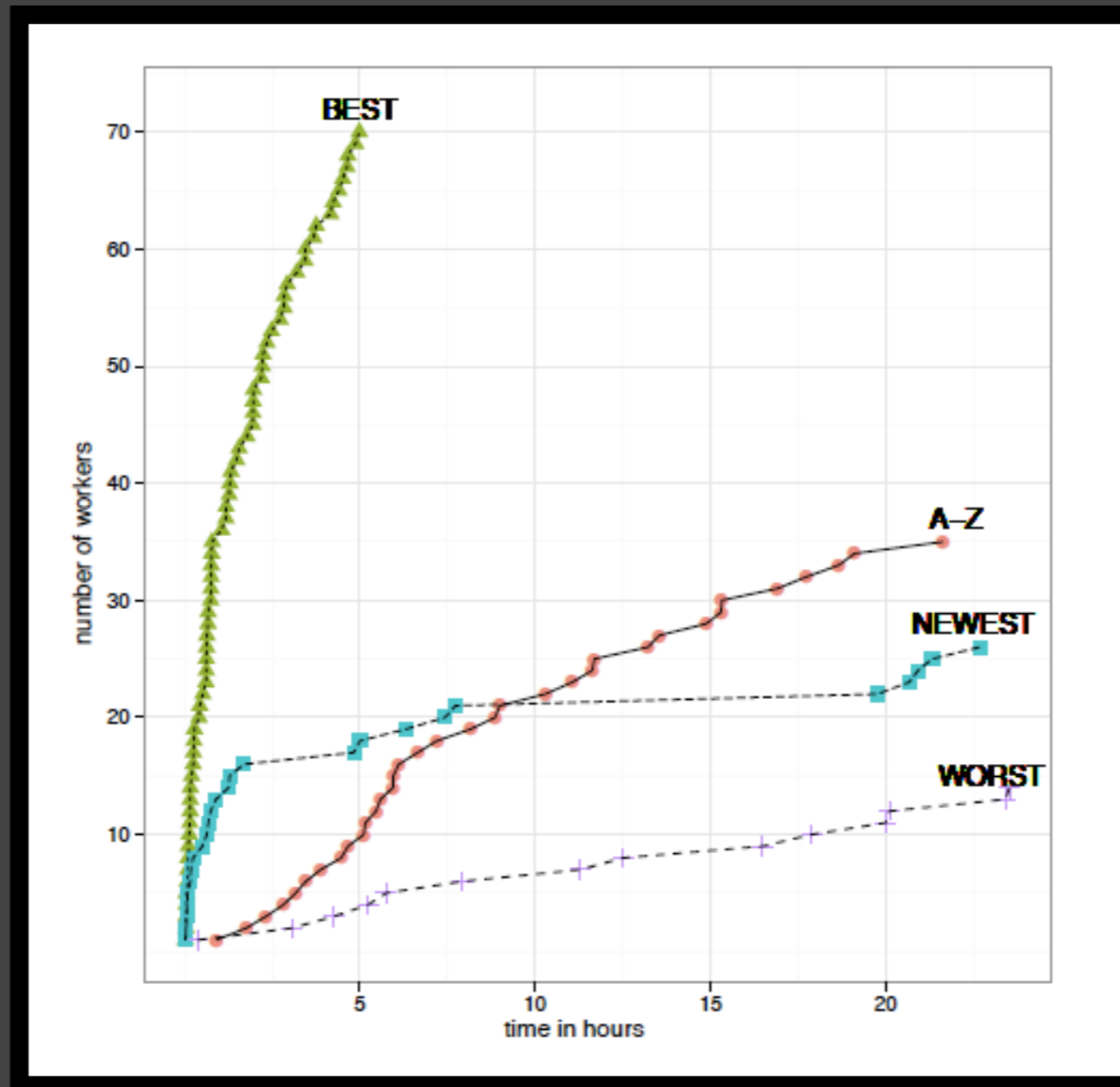
PULL METHODS

Pull Methods

workers  system

The system merely sets up the environment to allow workers to assign themselves (or each other) tasks.

Search locating tasks



(Chilton et al., 2010)



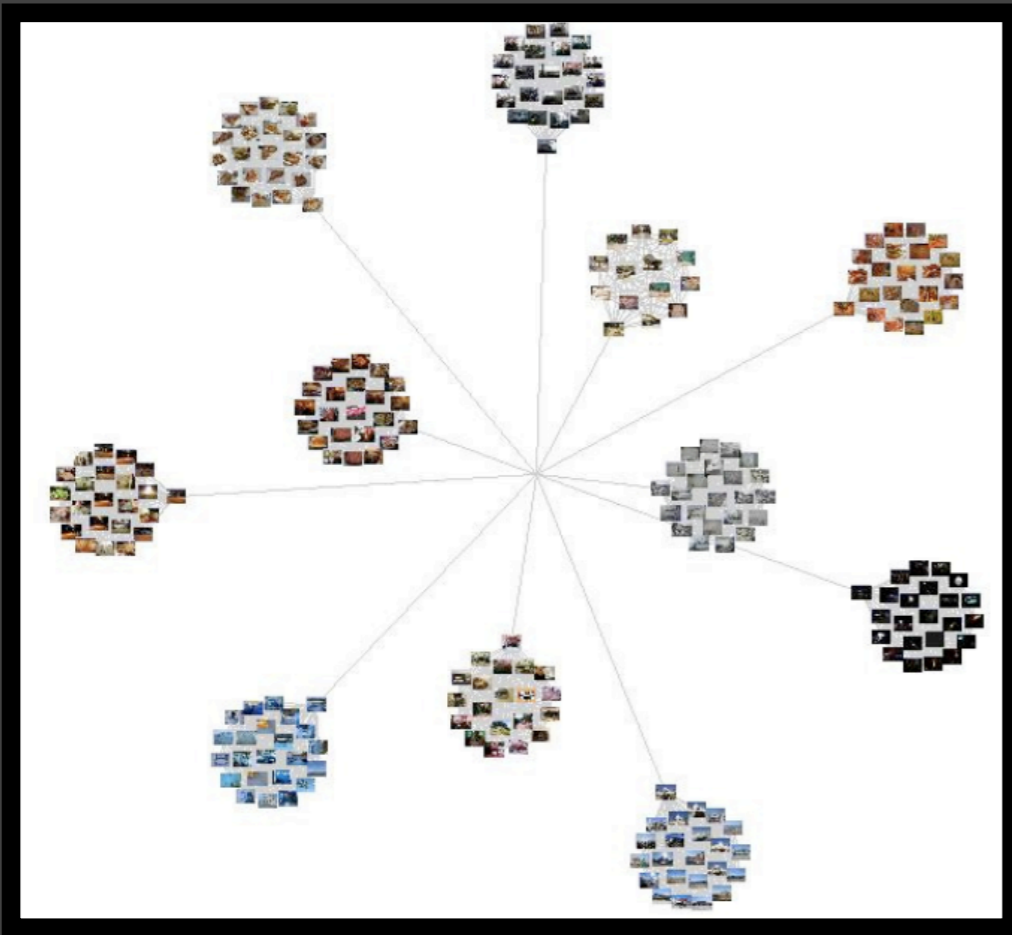
Visualization

locating particular input objects



Visualization

locating particular input objects



(Borden, 2006)



Task Recommendation personalization



Task Recommendation personalization

Content-Based

find similarities between worker profile and task characteristics.

Collaborative Filtering

make use of preference information about tasks (e.g., ratings) to infer similarities between workers.

Hybrid

a mix of content-based and collaborative filtering methods.



Task Recommendation personalization

Content-Based

find similarities between worker profile and task characteristics.

Collaborative Filtering

make use of preference information about tasks (e.g., ratings) to infer similarities between workers.

Hybrid

a mix of content-based and collaborative filtering methods.



Articles you might like to edit, from SuggestBot

SuggestBot predicts that you will enjoy editing some of these articles. Have fun!

Stubs	Cleanup
Jamie Waylett	Devon Murray
Phineas Nigellus	✓ David Heyman
Christian Coulson	Chris Rankin
Molly Parker	
Alfred Enoch	Merge
✓ Hog's Head	Scabbers
Harry Potter and the Half-Blood Prince (film)	Obliviator
Erskine William Gladstone	Minor Dark wizards in Harry Potter
Dartmoor Preservation Association	Add Sources
Unauthorized Chinese Harry Potter books	Tom Felton
Hogwarts headache	Filius Flitwick
Major powers - France	Ralph Fiennes
Biblical judges	Wikify
Manitoba Lotteries Corp.	✓ Derren Litten
Madam Hooch	Shambuka
Adrian Rawlins	Theatre in education
Geraldine Somerville	Expand
✓ Ottery St Catchpole	Strabag
Marvolo Gaunt	Froogle
	The Road Ahead

SuggestBot picks articles in a number of ways, from comparing articles that need work to other articles you've edited, to choosing articles randomly (ensuring that all articles with cleanup tags get a chance to be cleaned up). It tries to recommend only articles that other Wikipedians have marked as needing work. Your contributions make Wikipedia better -- thanks for helping.

If you have feedback on how to make SuggestBot better, please tell me on [SuggestBot's talk page](#). Thanks from [ForteTuba](#), SuggestBot's caretaker.

P.S. You received these suggestions because your name was listed on the [SuggestBot request page](#). If this was in error, sorry about the confusion. SuggestBot 04:05, 20 March 2006

SuggestBot (Cosley et al., 2006)



Peer Routing

people's knowledge of each other



Peer Routing

people's knowledge of each other



DARPA Red Balloon Challenge

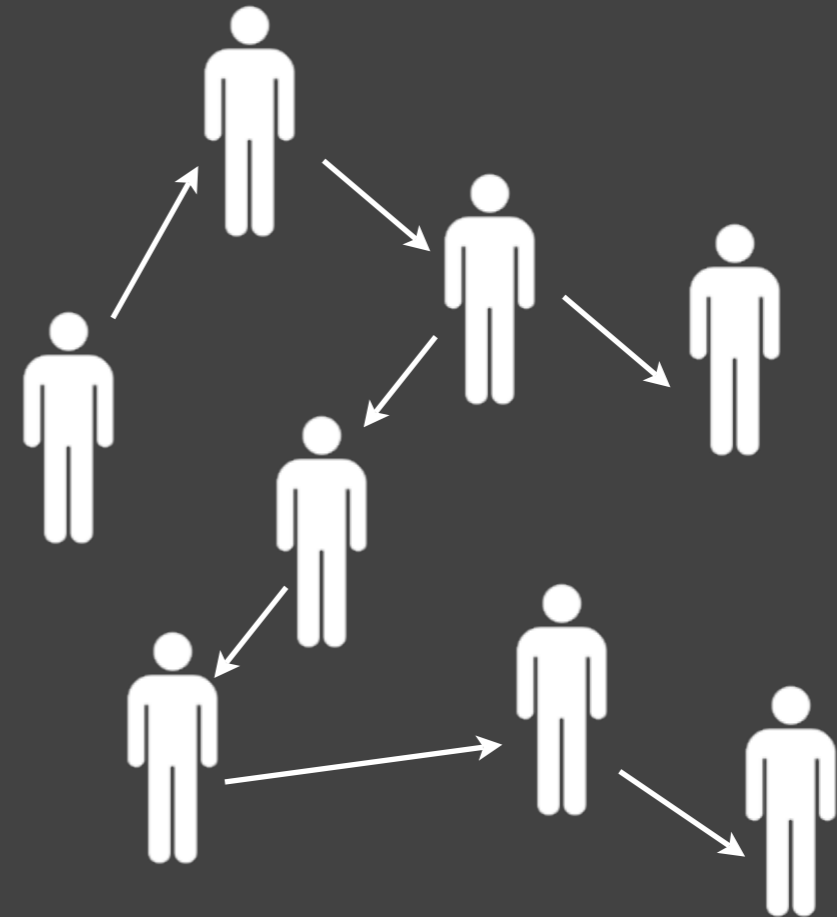


Peer Routing

people's knowledge of each other



DARPA Red Balloon Challenge



(Zhang et al., 2011)



RECAP

TAKE-HOME

“Wisdom of the **individuals** in the crowd”

VII

CONCLUSION

Conclusion

Summary

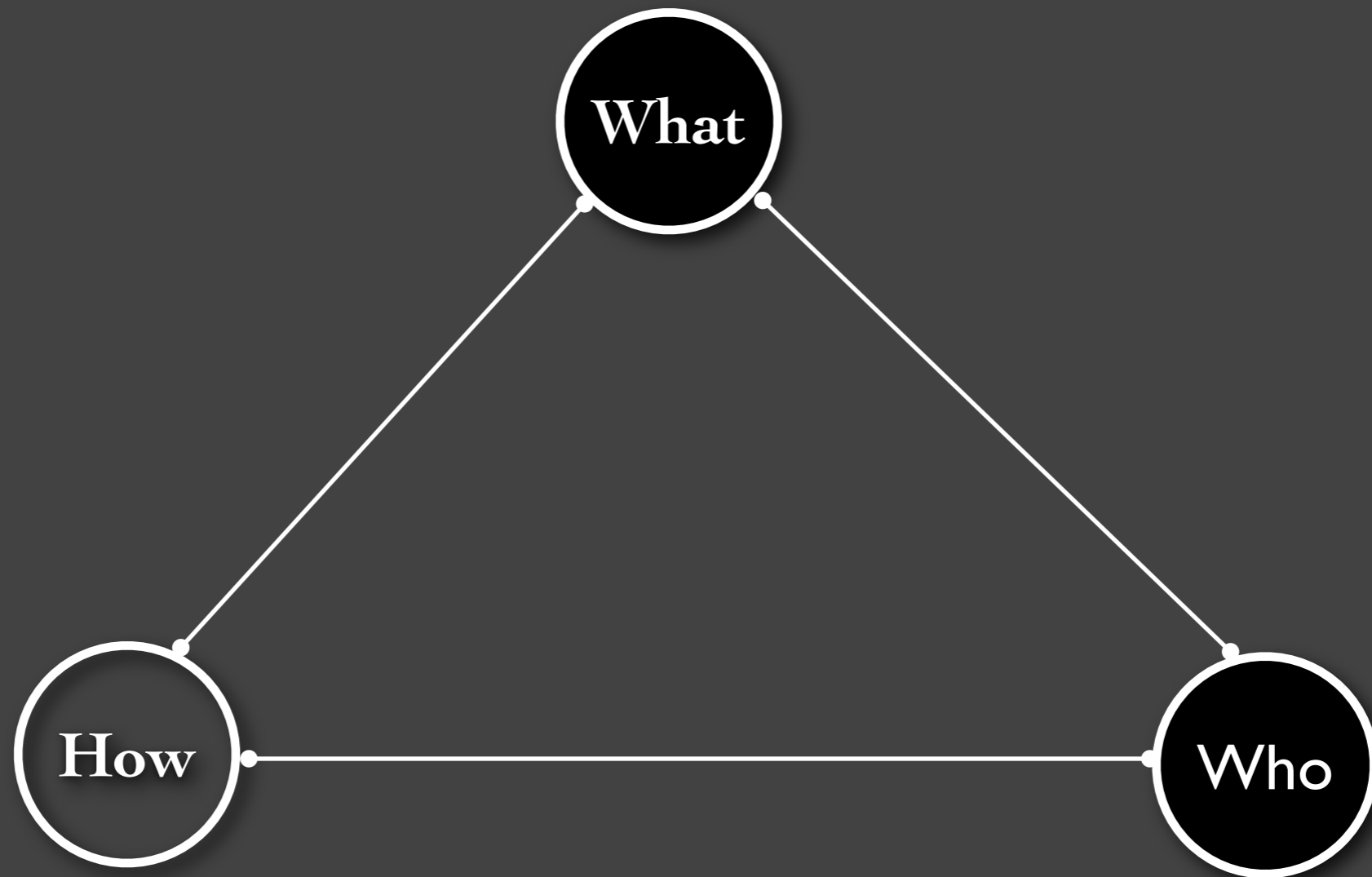
What have we learned?

Closing

What are some opportunities for AI research?

SUMMARY

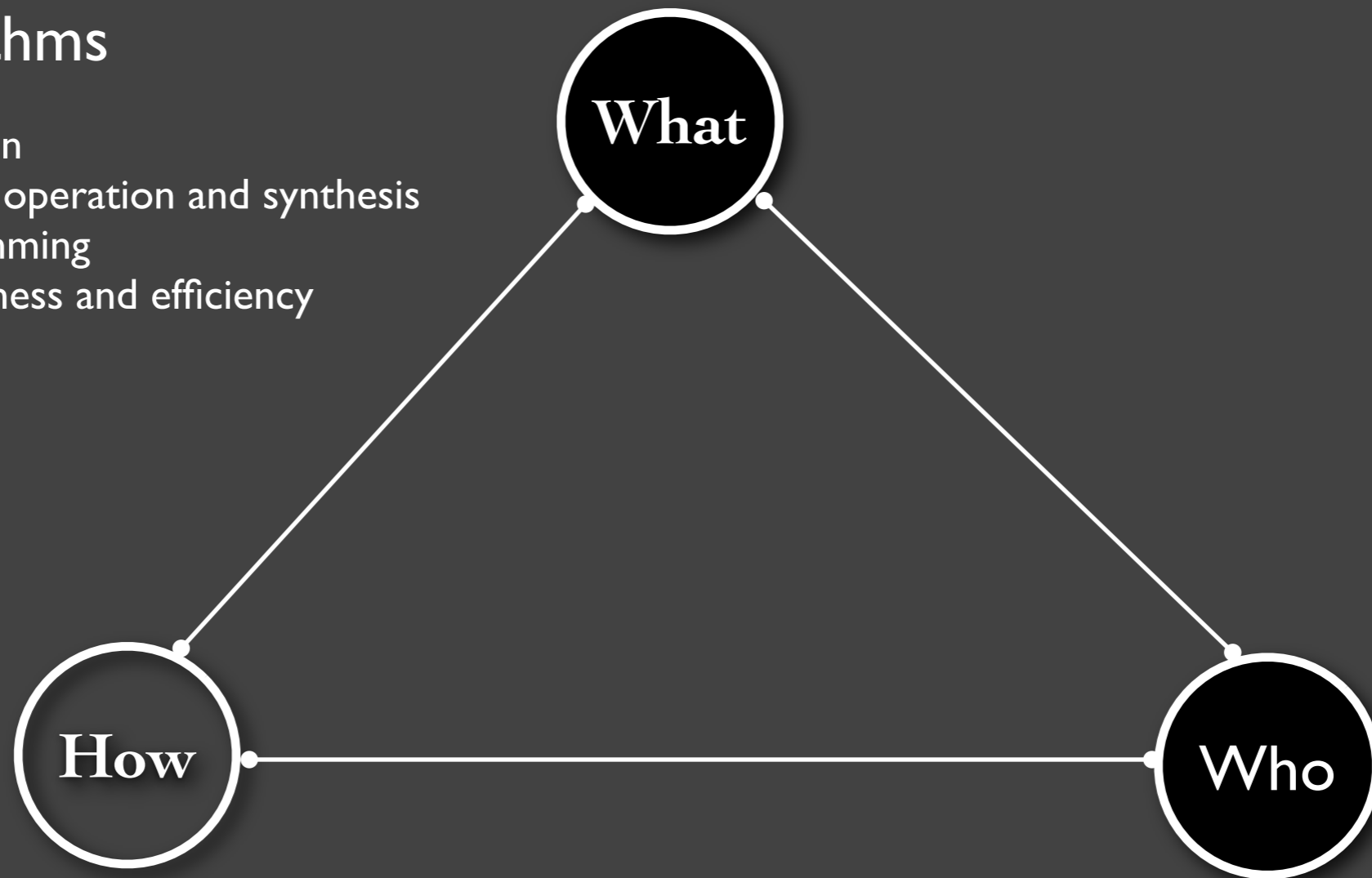
Bird's Eye View of this tutorial



Bird's Eye View of this tutorial

Human Computation Algorithms

- definition
- control, operation and synthesis
- programming
- correctness and efficiency



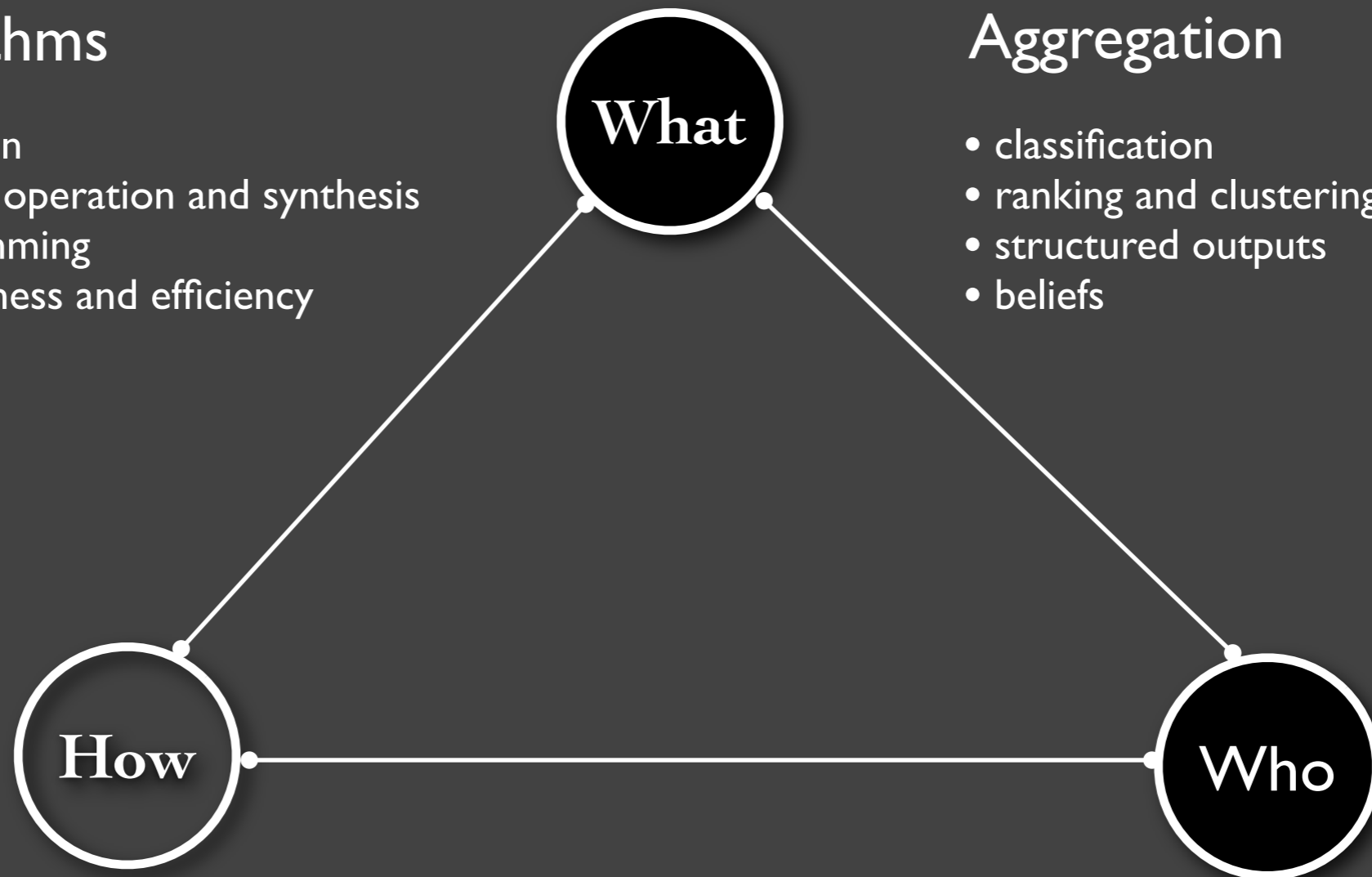
Bird's Eye View of this tutorial

Human Computation Algorithms

- definition
- control, operation and synthesis
- programming
- correctness and efficiency

Output Aggregation

- classification
- ranking and clustering
- structured outputs
- beliefs



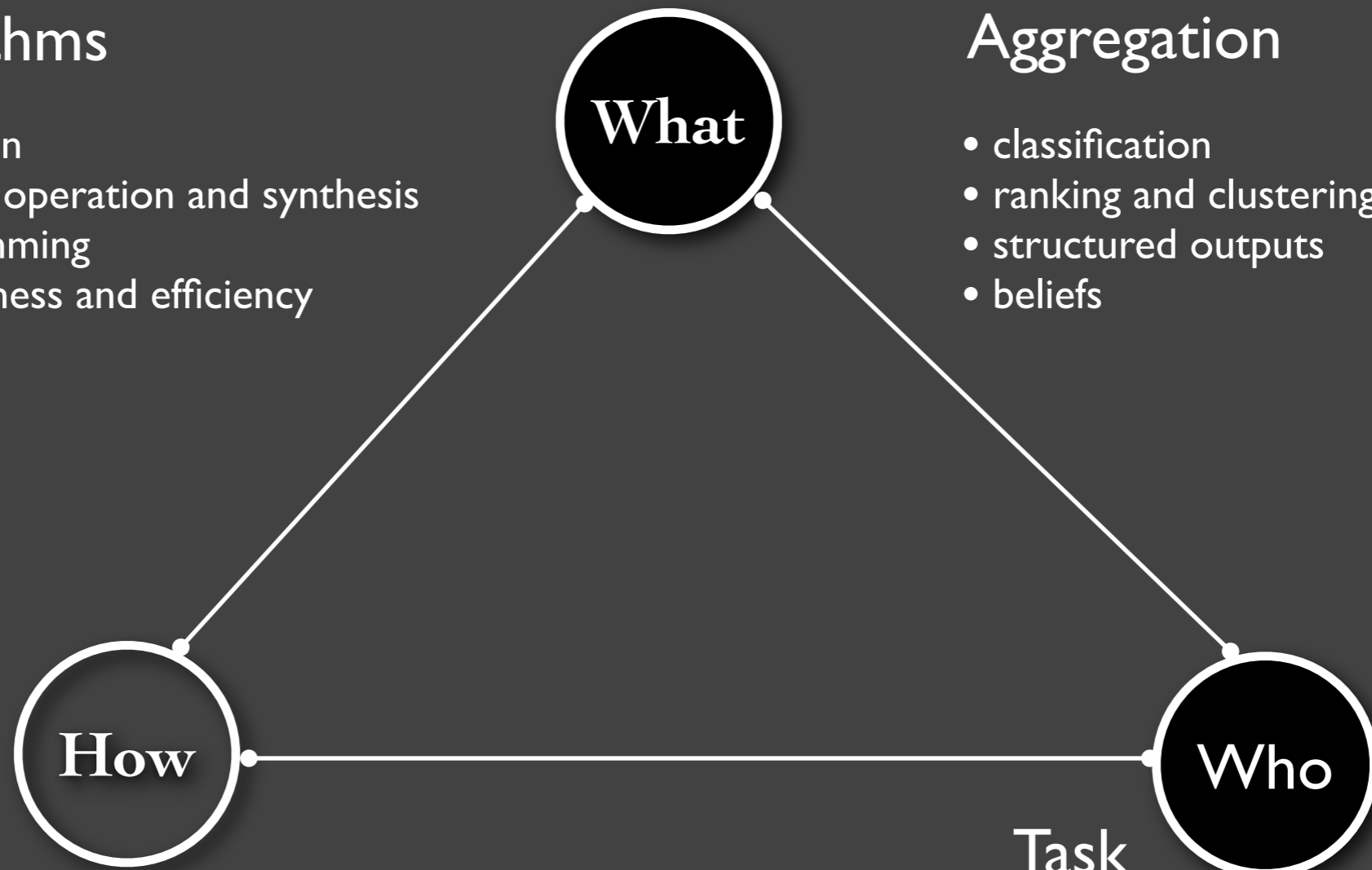
Bird's Eye View of this tutorial

Human Computation Algorithms

- definition
- control, operation and synthesis
- programming
- correctness and efficiency

Output Aggregation

- classification
- ranking and clustering
- structured outputs
- beliefs



Task Routing

- push versus pull
- allocation / matching
- inference / online learning

Bird's Eye View of this tutorial

Human Computation Algorithms

- definition
- control, operation and synthesis
- programming
- correctness and efficiency

Output Aggregation

- classification
- ranking and clustering
- structured outputs
- beliefs

What

How

Who

Designing for Human Computers

- who they are
- what are their wants and needs

The Art of Asking Questions

- task design
- game design

Task Routing

- push versus pull
- allocation / matching
- inference / online learning

CLOSING

The Role of AI in human computation



The Role of AI in human computation

AI as requesters

learning to recognize objects, translate sentences, classifying music by querying human teachers.



The Role of AI in human computation

AI as requesters

learning to recognize objects, translate sentences, classifying music by querying human teachers.

AI as optimizers

improve the accuracy and efficiency of human computation algorithms.



The Role of AI in human computation

AI as requesters

learning to recognize objects, translate sentences, classifying music by querying human teachers.

AI as optimizers

improve the accuracy and efficiency of human computation algorithms.

AI as enablers

make human computers better, e.g., by organizing and displaying information to workers.



The Role of AI in human computation

AI as requesters

learning to recognize objects, translate sentences, classifying music by querying human teachers.

AI as optimizers

improve the accuracy and efficiency of human computation algorithms.

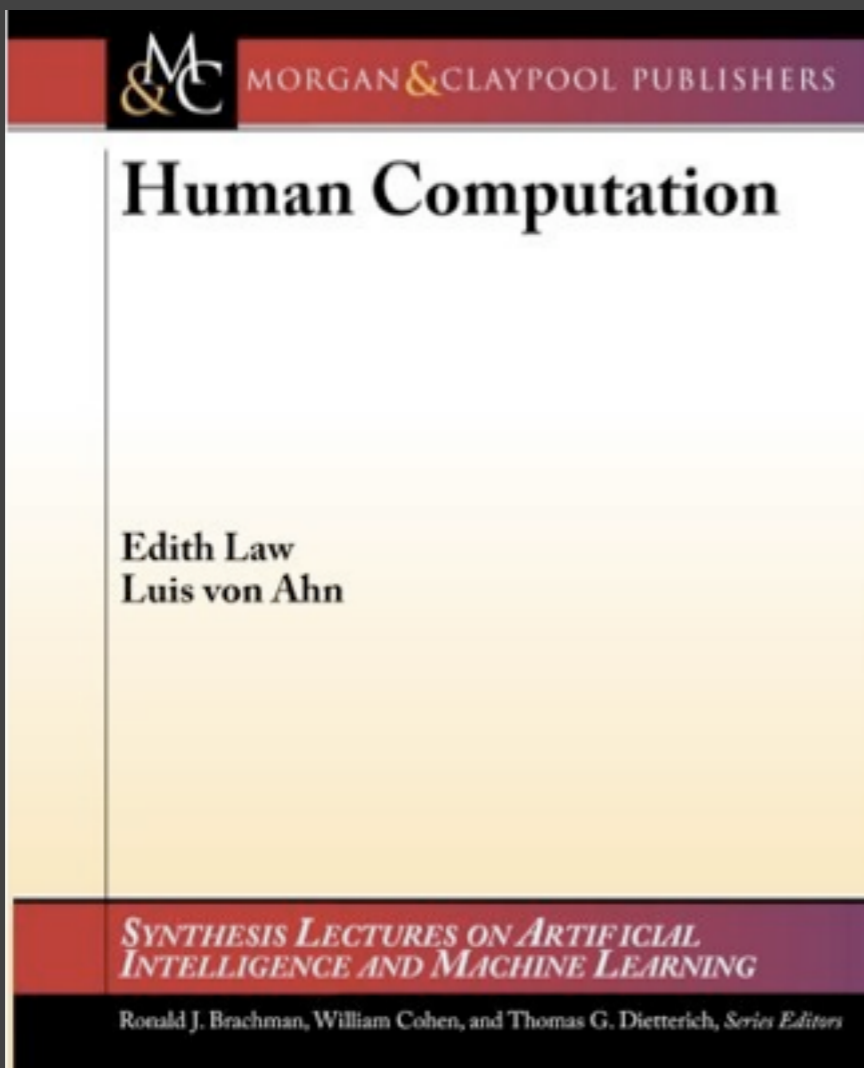
AI as enablers

make human computers better, e.g., by organizing and displaying information to workers.

AI as workers

perform tasks that they are better at than humans.






a conceptual framework
an annotated bibliography
a place to get ideas for research
a work in progress

free for you! Come pick one up
during the break.

Other resources:

<http://humancomputation.com/book>



**THANK YOU &
CATCH YOU @ COFFEE!**