Web Science

12 February 2019

Maria Maistro and Christina Lioma
{mm, c.lioma}@di.ku.dk

Lecture plan

- Collective Intelligence
- Crowdsourcing

Let’s start with an experiment

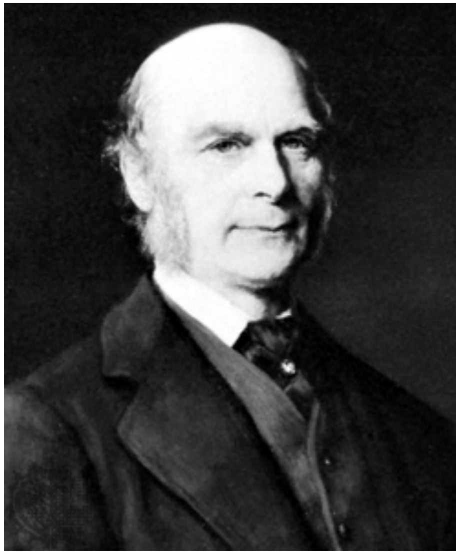
Let’s start with an experiment

How many M&Ms are in this bag?

https://goo.gl/forms/oWowMNJdOAPD2O6u1

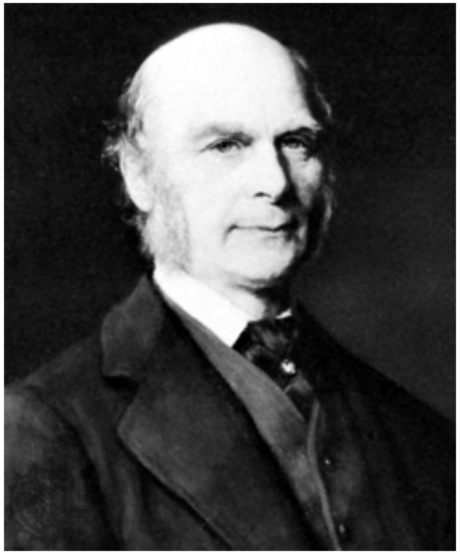
https://goo.gl/forms/Bp71901mX8d9x7vC2

Sir Francis Galton, 1822 − 1911, Victorian polymath



Charles Darwin's half-cousin

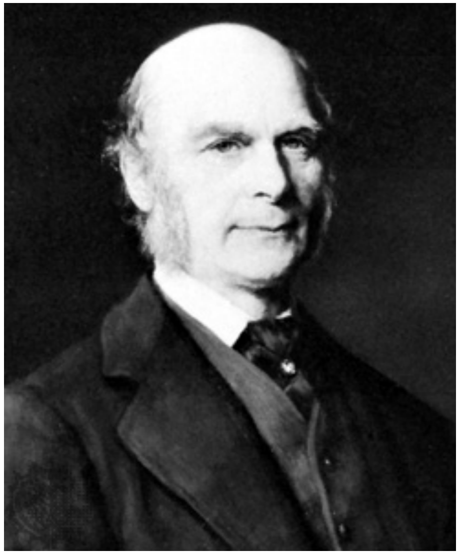
Sir Francis Galton, 1822 – 1911, Victorian polymath

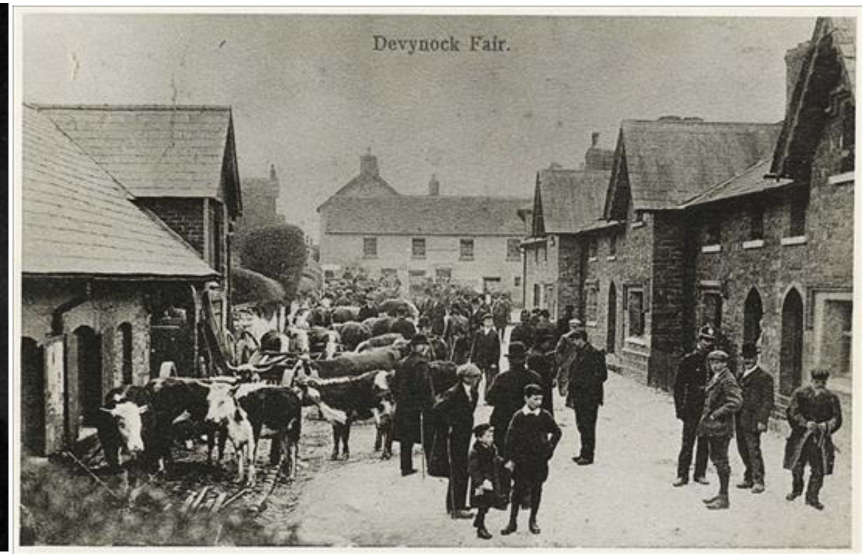


Charles Darwin’s half-cousin

- Created statistical concept of correlation
- Introduced questionnaires & surveys
- Coined terms “eugenics”, “nature versus nurture”
- Founded psychometrics & differential psychology
- Invented fingerprint classification in forensics
- Co-founded scientific meteorology, first weather map

Sir Francis Galton, Cambridge University 1906

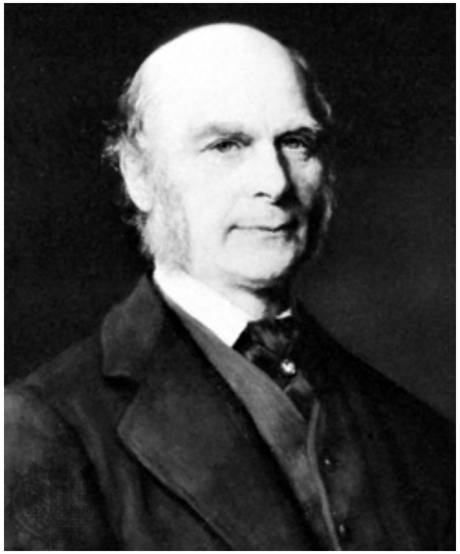


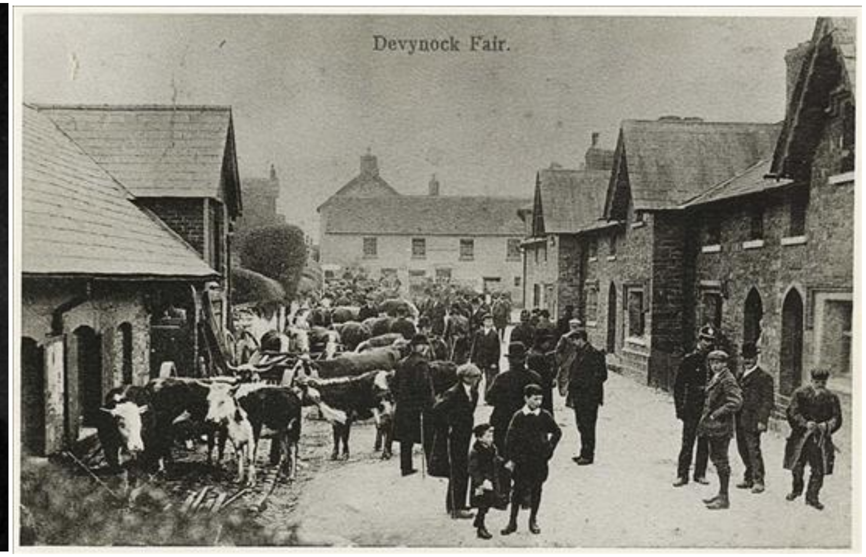


Guess the weight of an ox − prize for the best guess

- People placed wagers on the ox’s weight

Sir Francis Galton, Cambridge University 1906

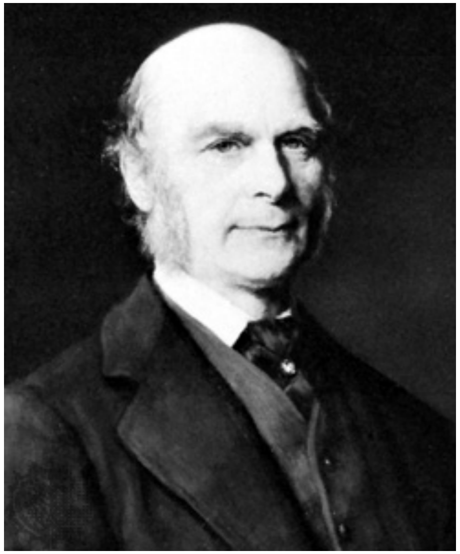


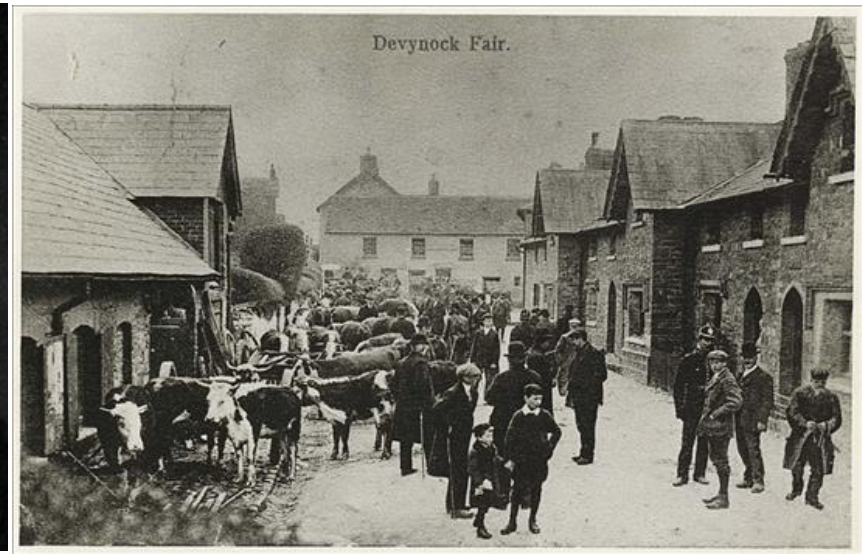


Guess the weight of an ox − prize for the best guess

- People placed wagers on the ox's weight
- Galton analysed 800 guesses
- Crowd's median guess: 542kg
- No one found the exact weight: 543kg

Sir Francis Galton, Cambridge University 1906





If you put together a big enough and diverse enough group of people and ask them to make a decision, that group's decision will, over time, be intellectually superior to the isolated individual

“Collective intelligence” (CI), Levy (philosopher) 1994

The group can be smart in a way that none of its members is

“Collective intelligence” (CI), Levy (philosopher) 1994

The group can be smart in a way that none of its members is

Intelligence definition: the ability to solve problems

- A system is more intelligent than another system if, in a given time interval, it can:
  - solve more problems, or
  - find better solutions to the same problems

“Collective intelligence” (CI), Levy (philosopher) 1994

The group can be smart in a way that none of its members is

Intelligence definition: the ability to solve problems

- A system is more intelligent than another system if, in a given time interval, it can:
  - solve more problems, or
  - find better solutions to the same problems

Collective Intelligence: a group can find more or better solutions than the whole of all solutions that would be found by its members working individually

Social insects

Complex & highly intelligent how:

- ants map out their environment
- termites build their mounds

Social insects

Complex & highly intelligent how:

- ants map out their environment
- termites build their mounds

Individually they are not intelligent (limited information processing capacities). But they are **many**.

Social insects

Complex & highly intelligent how:

- ants map out their environment
- termites build their mounds

Individually they are not intelligent (limited information processing capacities). But they are **many**.

Self-organising system: very large numbers of simple components interacting locally to produce global organization and adaptation.

Termites building mounds:

- At first, different termites drop mud randomly

Termites building mounds:

- At first, different termites drop mud randomly
- The presence of a mud heap incites other termites to add mud to that heap, rather than start a new heap

Termites building mounds:

- At first, different termites drop mud randomly
- The presence of a mud heap incites other termites to add mud to that heap, rather than start a new heap
- The larger the heap, the more attractive it is to other termites

Termites building mounds:

- At first, different termites drop mud randomly
- The presence of a mud heap incites other termites to add mud to that heap, rather than start a new heap
- The larger the heap, the more attractive it is to other termites
- Small heaps will be abandoned
- Large heaps will grow into tall columns

Termites building mounds:

- At first, different termites drop mud randomly
- The presence of a mud heap incites other termites to add mud to that heap, rather than start a new heap
- The larger the heap, the more attractive it is to other termites
- Small heaps will be abandoned
- Large heaps will grow into tall columns
- Columns will grow towards each other until they touch → arch

Termites building mounds:

- At first, different termites drop mud randomly
- The presence of a mud heap incites other termites to add mud to that heap, rather than start a new heap
- The larger the heap, the more attractive it is to other termites
- Small heaps will be abandoned
- Large heaps will grow into tall columns
- Columns will grow towards each other until they touch → arch
- Arch will grow until it touches other arches → intricate structure of interlocking arches

Swarm intelligence (in AI)

Emergent collective intelligence of groups of simple agents

The key is that complex behaviour (problem solving) arises out of a composition of simple behaviours

Ant colonies, bird flocking, animal herding, microbial intelligence, etc.

Six Characteristics of Collective Intelligence:

Six Characteristics of Collective Intelligence:

1. Group of individuals (e.g., people, robots, software agents, or a mixture of these)

Six Characteristics of Collective Intelligence:

1. Group of individuals (e.g., people, robots, software agents, or a mixture of these)
2. Big enough group, diverse enough group

Six Characteristics of Collective Intelligence:

1. Group of individuals (e.g., people, robots, software agents, or a mixture of these)
2. Big enough group, diverse enough group
3. Local and often random behaviours

Six Characteristics of Collective Intelligence:

1. Group of individuals (e.g., people, robots, software agents, or a mixture of these)
2. Big enough group, diverse enough group
3. Local and often random behaviours
4. Simple interactions, often following simple rules

Six Characteristics of Collective Intelligence:

1. Group of individuals (e.g., people, robots, software agents, or a mixture of these)
2. Big enough group, diverse enough group
3. Local and often random behaviours
4. Simple interactions, often following simple rules
5. Self-organising over time

Six Characteristics of Collective Intelligence:

1. Group of individuals (e.g., people, robots, software agents, or a mixture of these)
2. Big enough group, diverse enough group
3. Local and often random behaviours
4. Simple interactions, often following simple rules
5. Self-organising over time
6. De-centralised (no dictating how to behave)

Six Characteristics of Collective Intelligence:

1. Group of individuals (e.g., people, robots, software agents, or a mixture of these)
2. Big enough group, diverse enough group
3. Local and often random behaviours
4. Simple interactions, often following simple rules
5. Self-organising over time
6. De-centralised (no dictating how to behave)

→ Globally intelligent: more or better solutions than if working individually

Six Characteristics of Collective Intelligence:

1. **Group** of individuals (e.g., people, robots, software agents, or a mixture of these)
2. **Big** enough group, **diverse** enough group
3. **Local** and often **random behaviours**
4. **Simple interactions**, often following simple rules
5. **Self-organising** over time
6. **De-centralised** (no dictating how to behave)

→ Globally intelligent: more or better solutions than if working individually

The World Wide Web

Collective intelligence

- Possibility to seamlessly integrate machine and human intelligence at large scale

Collective intelligence

- Possibility to seamlessly integrate machine and human intelligence at large scale
- One of the holy grails of AI, known as **mixed-initiative systems** (Horvitz 2007)
  - Intelligent systems that can *collaborate* effectively & naturally with humans

Collective intelligence

- Possibility to seamlessly integrate machine and human intelligence at large scale
- One of the holy grails of AI, known as **mixed-initiative systems** (Horvitz 2007)
  - Intelligent systems that can *collaborate* effectively & naturally with humans (reasoning, communication, planning, execution, learning)

Collective intelligence

- Possibility to seamlessly integrate machine and human intelligence at large scale
- One of the holy grails of AI, known as **mixed-initiative systems** (Horvitz 2007)
  - Intelligent systems that can collaborate effectively & naturally with humans (reasoning, communication, planning, execution, learning)

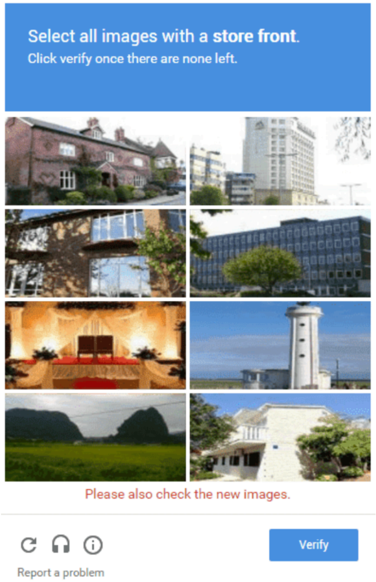
Crowdsourcing (type of Collective Intelligence)

- Taking a job traditionally done by a designated person and outsourcing it to an undefined, large group of people

Crowdsourcing: human-based computation

Use humans as processors in a distributed system

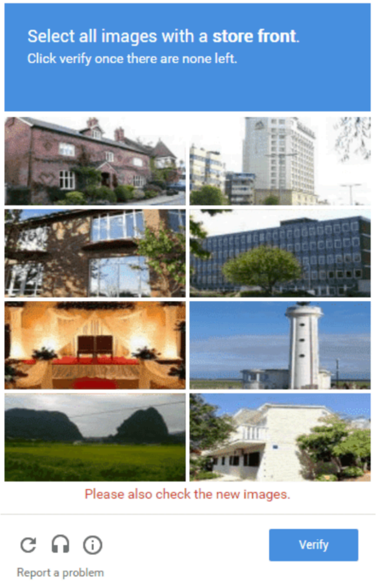
- Wikipedia, Captcha



Crowdsourcing: human-based computation

Use humans as processors in a distributed system

- Wikipedia, Captcha



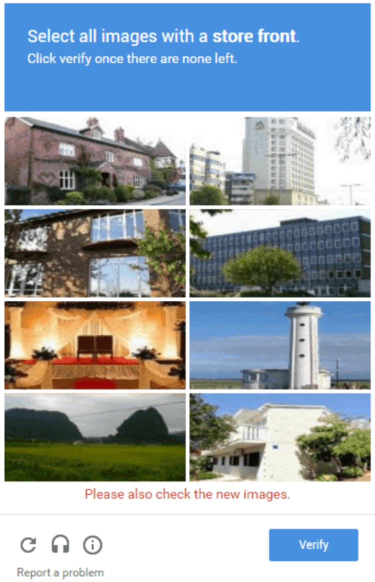
Amazon Mechanical Turk (mturk.com):

- Crowdsourcing platform

Crowdsourcing: human-based computation

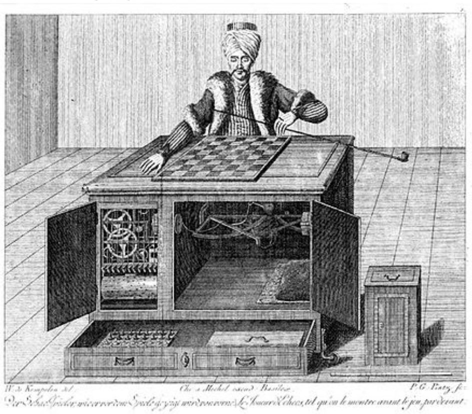
Use humans as processors in a distributed system.

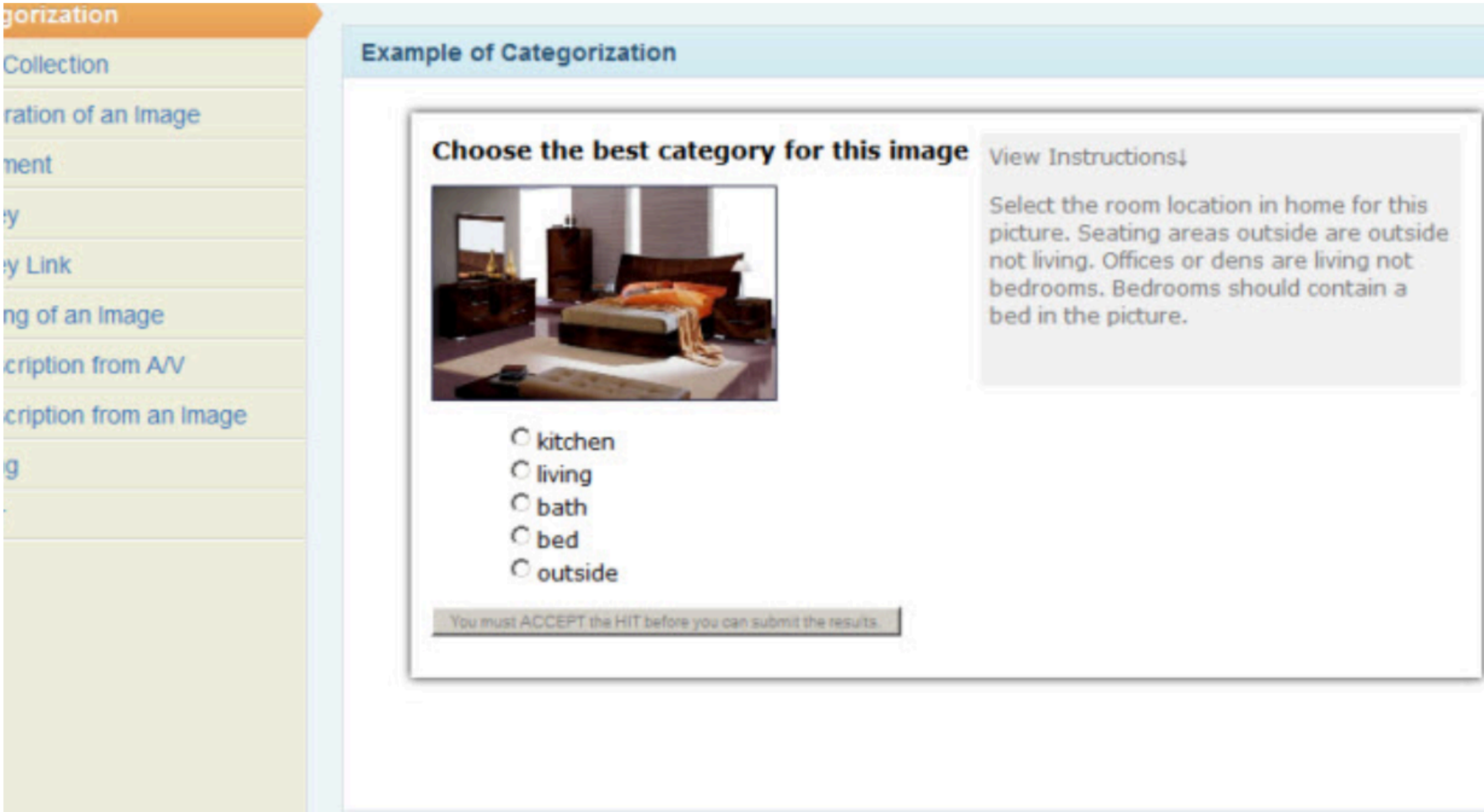
- Wikipedia, Captcha



Amazon Mechanical Turk (mturk.com):

- Crowdsourcing platform
- Named after Mechanical Turk, fake chess playing machine in 18C
- Artificial artificial intelligence




gorization
Collection
ration of an Image
ment
y
y Link
ng of an Image
cription from A/V
cription from an Image
g
Example of Categorization
Choose the best category for this image
View Instructions↓
Select the room location in home for this picture. Seating areas outside are outside not living. Offices or dens are living not bedrooms. Bedrooms should contain a bed in the picture.
kitchen
living
bath
bed
outside
You must ACCEPT the HIT before you can submit the results.

Amazon Mechanical Turk

- <u>Requesters</u>

- <u>Workers</u>

Amazon Mechanical Turk

- Requesters create human intelligence tasks (HITs) via web services API or dashboards
- Workers

Amazon Mechanical Turk

- Requesters create human intelligence tasks (HITs) via web services API or dashboards
- Workers log in, choose HITs, perform them

Amazon Mechanical Turk

- Requesters create human intelligence tasks (HITs) via web services API or dashboards
- Workers log in, choose HITs, perform them
- Requesters assess results, pay per HIT satisfactorily completed

Amazon Mechanical Turk

- Requesters create human intelligence tasks (HITs) via web services API or dashboards
- Workers log in, choose HITs, perform them
- Requesters assess results, pay per HIT satisfactorily completed
- Currently >200,000 workers from >100 countries; millions of HITs completed

Crowdsourcing

- process for obtaining work or knowledge from large crowds through a system of invitations and incentives

Crowdsourcing

- process for obtaining work or knowledge from large crowds through a system of invitations and incentives
- a requestor (with some internal objective) solicits a group of workers to perform tasks in service of that objective

Crowdsourcing

- process for obtaining work or knowledge from large crowds through a system of invitations and incentives
- a requestor (with some **internal objective**) solicits a group of workers to perform tasks in service of that objective

Requestor's objective → utility function to be maximised

Example: obtain labels for a set of images

Requestor's objective includes:

- Maximising average quality of labels

Example: obtain labels for a set of images

Requestor’s objective includes:

- Maximising average quality of labels
- Minimising the number of tasks given to workers (to reduce own cost and/or worker burden)
- Minimising latency (get answers quickly)

Example: obtain labels for a set of images

Requestor's objective includes:

- Maximising average quality of labels
- Minimising the number of tasks given to workers (to reduce own cost and/or worker burden)
- Minimising latency (get answers quickly)

Requestor must choose:

- How many workers should be given a labelling task
- How their responses should be aggregated to estimate the best answer

Assumptions:

- workers act independently, interacting only through shared tasks
- Each worker has own utility function, which is often different from the collective's utility function

Assumptions:

- workers act independently, interacting only through shared tasks
- Each worker has own utility function, which is often different from the collective's utility function

Crowdsourcing for AI: use crowdsourcing to label training sets as input for data-hungry supervised algorithms

Advantages:

- wisdom of crowds
- round the clock availability of people
- low cost

Advantages:

- wisdom of crowds
- round the clock availability of people
- low cost

Challenges:

- Highly varying skills, abilities, motivation of workers

Advantages:

- wisdom of crowds
- round the clock availability of people
- low cost

Challenges:

- Highly varying skills, abilities, motivation of workers
- Different tasks may be individually easier or more difficult, requiring less or more work (iterations) on them

Methods to address these challenges:

1. Learning algorithms that model the accuracy of crowd workers

Methods to address these challenges:

1. Learning algorithms that model the accuracy of crowd workers
2. Aggregation methods for predicting true answers from error-prone and disagreeing workers

Methods to address these challenges:

1. Learning algorithms that model the accuracy of crowd workers
2. Aggregation methods for predicting true answers from error-prone and disagreeing workers
3. Control algorithms that choose which tasks to request and which individuals should work on them

Methods to address these challenges:

1. Learning algorithms that model the accuracy of crowd workers
2. Aggregation methods for predicting true answers from error-prone and disagreeing workers
3. Control algorithms that choose which tasks to request and which individuals should work on them

Crowdsourcing quality: highly dependent on 1-3

Crowdsourcing strategies

- Post hoc response aggregation (after responses have been received)

- Dynamically fit tasks to workers (while responses are being received)

Given:

- set of questions, workers, and their responses (some questions are answered by more than one worker)

Given:

- set of questions, workers, and their responses (some questions are answered by more than one worker)
- questions are objective (each has a unique answer)

Given:

- set of questions, workers, and their responses (some questions are answered by more than one worker)
- questions are objective (each has a unique answer)
- workers may not answer correctly
- majority of workers are more likely to answer correctly than not

Given:

- set of questions, workers, and their responses (some questions are answered by more than one worker)
- questions are objective (each has a unique answer)
- workers may not answer correctly
- majority of workers are more likely to answer correctly than not

Goal:

1) exploit redundancy
2) learn and track worker skills

1. Exploit redundancy by comparing different workers responses to the same question

- Snow et al 2008: with simple majority voting a crowd of novices outperformed an expert on sentiment analysis and semantic word similarity

1. Exploit redundancy by comparing different workers responses to the same question

- Snow et al 2008: with simple majority voting a crowd of novices outperformed an expert on sentiment analysis and semantic word similarity

2. Learn and track the skills of workers

- Instead of majority voting, weigh worker responses by using models of workers' abilities. E.g. if a worker is excellent at translating French to English, assume that their English to French translations are of high quality

Improve on majority voting by modelling worker skill

- Supervised learning: give workers questions for which gold standard answers are already known
- If worker fails to correctly answer: dismiss or have their weights lowered

Improve on majority voting by modelling worker skill

- Supervised learning: give workers questions for which gold standard answers are already known
- If worker fails to correctly answer: dismiss or have their weights lowered

Common problem: **gaming behaviour**

- answer the first few questions and then decrease effort and accuracy on remaining questions

Solution: intermix questions with known and unknown answers

Early use of expectation maximisation

David & Skene 1979 (medical diagnosis)

- There is a single question with an unknown correct answer
- $P_w(r|a)$: probability that worker w will give response r when true answer is a
- If $r \neq a$, an expert worker would have $P_w(r|a) = 0$
- Worker responses are conditionally independent of each other, given the true answer
- Expectation maximisation: iterative algorithm to estimate which answers are correct at the same time that the algorithm learns the model of worker accuracies

Expectation Maximisation

- Start by taking a majority vote and use that to determine an initial guess of the correct answer to each question

Expectation Maximisation

- Start by taking a majority vote and use that to determine an initial guess of the correct answer to each question
- Score each worker based on how many answers he/she got right

Expectation Maximisation

- Start by taking a majority vote and use that to determine an initial guess of the correct answer to each question
- Score each worker based on how many answers he/she got right
- Weigh each worker's votes based on their score (better workers count more)

Expectation Maximisation

- Start by taking a majority vote and use that to determine an initial guess of the correct answer to each question
- Score each worker based on how many answers he/ she got right
- Weigh each worker's votes based on their score (better workers count more)
- Since weighted votes likely produce a different set of correct answers, recompute each worker's score

Expectation Maximisation

- Start by taking a majority vote and use that to determine an initial guess of the correct answer to each question
- Score each worker based on how many answers he/she got right
- Weigh each worker’s votes based on their score (better workers count more)
- Since weighted votes likely produce a different set of correct answers, recompute each worker’s score
- Repeat until quiescence

Expectation Maximisation:

Expectation: given estimates of all the probabilities, compute the probability of each possible answer using Bayes rules and assuming conditional independence

Maximisation: given the posterior probability of each possible answer, compute new parameters than maximise the likelihood of each voter’s response

Expectation Maximisation:

<u>Expectation:</u> given estimates of all the probabilities, compute the probability of each possible answer using Bayes rules and assuming conditional independence

<u>Maximisation:</u> given the posterior probability of each possible answer, compute new parameters than maximise the likelihood of each voter’s response

- assigns higher weights to good workers

Expectation Maximisation:

<u>Expectation:</u> given estimates of all the probabilities, compute the probability of each possible answer using Bayes rules and assuming conditional independence

<u>Maximisation:</u> given the posterior probability of each possible answer, compute new parameters than maximise the likelihood of each voter's response

- assigns higher weights to good workers
- allows for single strong worker to overrule multiple weak workers

Expectation Maximisation:

Expectation: given estimates of all the probabilities, compute the probability of each possible answer using Bayes rules and assuming conditional independence

Maximisation: given the posterior probability of each possible answer, compute new parameters than maximise the likelihood of each voter's response

- assigns higher weights to good workers
- allows for single strong worker to overrule multiple weak workers
- predicted answer may not always be the majority vote

EM extension (Whitehill et al. 2009)

Let on average students have an 80% chance of correctly answering questions. → Jane has 80% chance of answering correctly Q13.

EM extension (Whitehill et al. 2009)

Let on average students have an 80% chance of correctly answering questions. → Jane has 80% chance of answering correctly Q13.

If everyone else answers Q13 wrong → Q13 is especially hard and we must lower Jane’s chances.

EM extension (Whitehill et al. 2009)

Let on average students have an 80% chance of correctly answering questions. → Jane has 80% chance of answering correctly Q13.

If everyone else answers Q13 wrong → Q13 is especially hard and we must lower Jane's chances.

Idea: use worker errors to update estimate of task difficulty and worker accuracy.

Same idea as in EM, but probability computations include worker expertise and task difficulty.

EM extension (Welinder et al. 2010)

- Questions have many features → Workers modelled as linear classifiers who respond by weighing those features

EM extension (Welinder et al. 2010)

- Questions have many features → Workers modelled as linear classifiers who respond by weighing those features
- Account not only for question difficulty and worker skills, but also for arbitrary question and worker features.

E.g. worker a can be good at identifying types of birds, but only when viewed from the back.

EM extension (Welinder et al. 2010)

- Questions have many features → Workers modelled as linear classifiers who respond by weighing those features
- Account not only for question difficulty and worker skills, but also for arbitrary question and worker features.

E.g. worker a can be good at identifying types of birds, but only when viewed from the back.

Question features need not be specified a priori: the algorithm learns the features.

Excellent performance but low interpretability

Further extensions (I)

Prelec and Seung 2007

Find correct answers missed by the majority by asking workers to predict coworkers mistakes

Further extensions (I)

Prelec and Seung 2007

Find correct answers missed by the majority by asking workers to predict coworkers mistakes

Worker is asked to

- endorse the response mostly likely to be true, and
- predict the proportion of the sample workers that will endorse each possible response.

Further extensions (I)

<u>Prelec and Seung 2007</u>

Find correct answers missed by the majority by asking workers to predict coworkers mistakes

Worker is asked to

- endorse the response mostly likely to be true, and
- predict the proportion of the sample workers that will endorse each possible response.

<u>Lin et al. 2012</u>

When requesters cannot enumerate all possible answers for the worker or when the solution space is infinitely large

Further extensions (II)

<u>Raykar et al. 2010</u>

Fully automatic:

- learn worker abilities
- infers correct responses
- learns a logistic regression classifier to predict future crowd responses or the answer (no need to consult human workers in the future)

Instead of only reconciling multiple responses, get the answers themselves

Further extensions (III)

<u>Dekel and Shamir 2009</u>

Limit influence of bad workers.

- use worker responses to train a SVM
- add constrains to the loss function such that no bad worker can overly influence the learned weights
- prune bad workers

Further extensions (III)

Dekel and Shamir 2009

Limit influence of bad workers.

- use worker responses to train a SVM
- add constrains to the loss function such that no bad worker can overly influence the learned weights
- prune bad workers

Wauthier and Jordan 2011

Relax the idea that tasks must contain a correct answer.

- predict each worker’s response to a future subjective question.