Washington State University

Near-Optimally Teaching the Crowd To Classify

Authors: Adish Singla, Ilija Bogunovic, Gabor Bartók, Amin Karbasi, and Andreas Krause

Proceedings: ICML 2014

Presented by: Gina Sprint 1/27/15

Author Information

Adish Singla

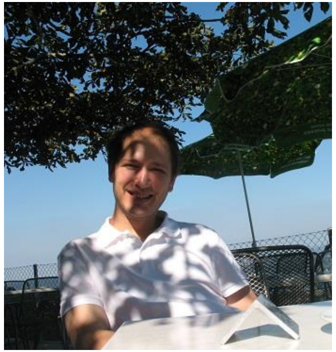
- PhD Student, ETH

Ilija Bogunovic

- PhD Student, EPFL

Gabor Bartok

- Postdoc, ETH
- Currently at Google Zurich



Amin Karbasi

- Postdoc, ETH
- Currently Assistant Professor at Yale

Andreas Krause

- Associate Professor, ETH

http://www.inf.ethz.ch/personal/adish.singla/, http://lions.epfl.ch/phdstudents/ilija.bogunovic, http://people.inf.ethz.ch/gbartok/, http://las.ethz.ch/krausea.html, http://seas.yale.edu/faculty-research/faculty-directory/amin-karbasi

Crowd + Outsourcing

- Soliciting contributions from a large group of people
  - Subdivide tedious work
  - "Simple for humans, hard for computers"
- Online communities
- Example: Labeling large sets of data

Examples of Crowdsourcing?

- Crowdvoting
  - Gathering opinions
- Crowdsearching
  - Finding lost items, pets, people, etc.
- Crowdfunding
  - Lots of small $ contributions
- Crowdtesting
  - Testing software
- Crowdwisdom
  - Gathering information

Safire, William (February 5, 2009). "On Language". New York Times Magazine.

Issues with Crowdsourcing?

Advantages	Disadvantages

• Mass intelligence
• Workers on demand
  – Little to no skills required
  – No benefits packages
• Affordable price
• Diverse projects
	• No worker expertise
	• Low credibility
	  – “Noisy” results
	  – Careless effort
	• Not suitable for complex tasks
	• Time required for management
	• Collaboration is difficult

A. Sorokin and D. Forsyth, “Utility data annotation with amazon mechanical turk,” *Urbana*, vol. 51, no. 61, p. 820, 2008, http://www.hongkiat.com/blog/what-is-crowdsourcing/

Research in Crowdsourcing

- Tackling noisy responses
- Estimating worker "reliability"
- Compensation
- Complex tasks
- Educating the crowd

Nilesh Dalvi, Anirban Dasgupta, Ravi Kumar, and Vibhor Rastogi. 2013. Aggregating crowdsourced binary ratings. In *Proceedings of the 22nd international conference on World Wide Web* (WWW '13).., Rion Snow, Brendan O'Connor, Daniel Jurafsky, and Andrew Y. Ng. 2008. Cheap and fast---but is it good?: evaluating non-expert annotations for natural language tasks. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing* (EMNLP '08)., http://www.blogging4jobs.com/work/crowdsourcing-is-the-future-of-performance-reviews/#UP2IwRYqhRczCTKl.97

Crowdteaching

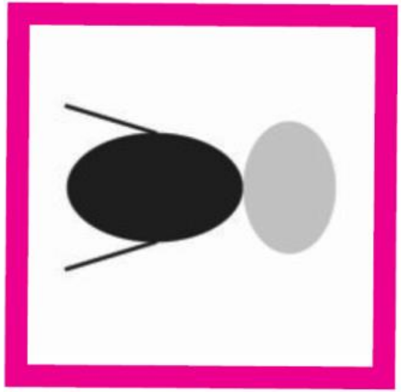
- “Can we teach workers in crowdsourcing services in order to improve their accuracy?”
- Example: Teaching the crowd to label images

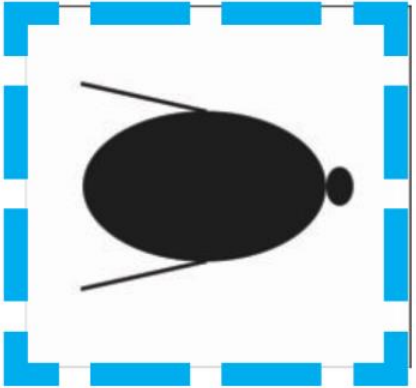
Classifying Animal Species

- Common in many projects (i.e. eBird)
- Classify an image as containing a butterfly or moth?
  - Butterfly → +
  - Moth → -

Classifying (Synthetic) Animal Species

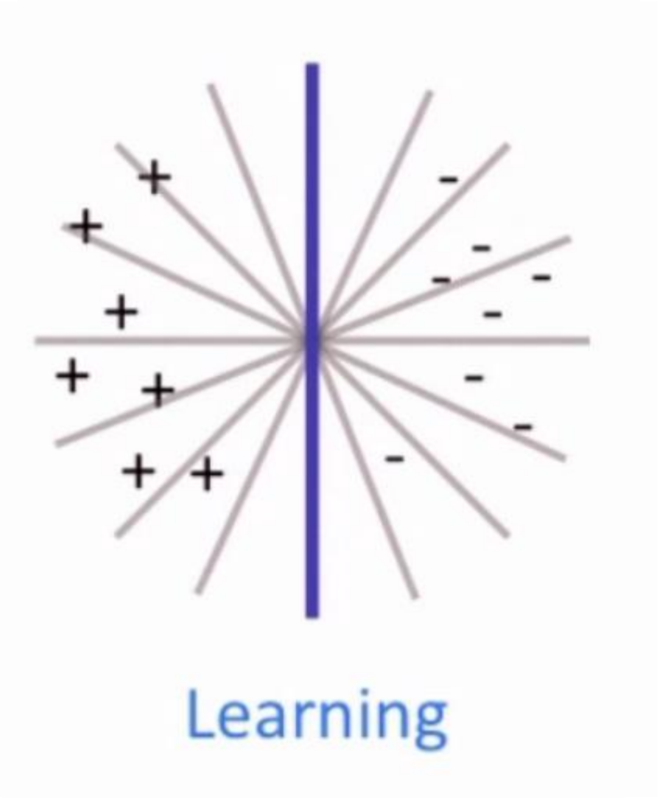
- Classify if an image contains a "Vespula" or "Weevil"
- Vespula → +
- Weevil → -
- How would you distinguish +/-?
  - f1: head/body size proportion
  - f2: body/head color contrast

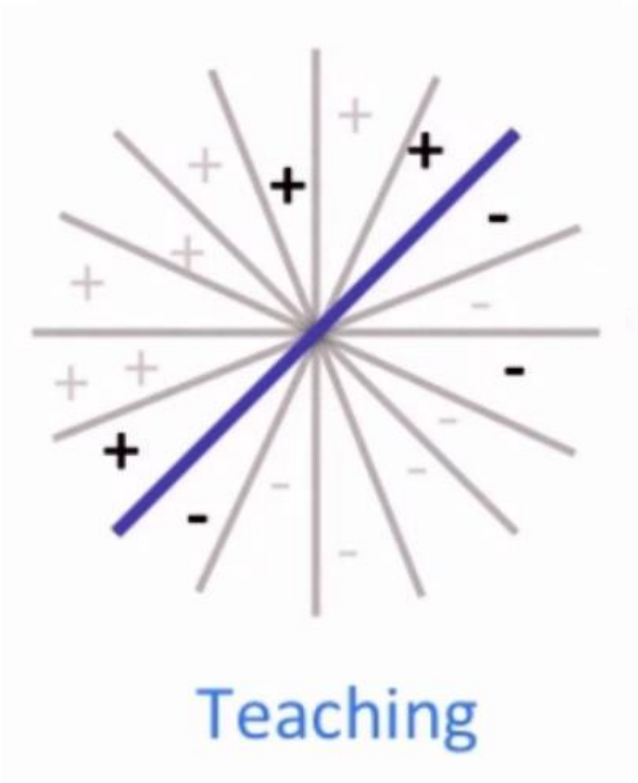




Vespula Weevil

Learning vs. Teaching





Challenge of the Teacher

How should we select training examples to teach the correct classification?

(as fast as possible)

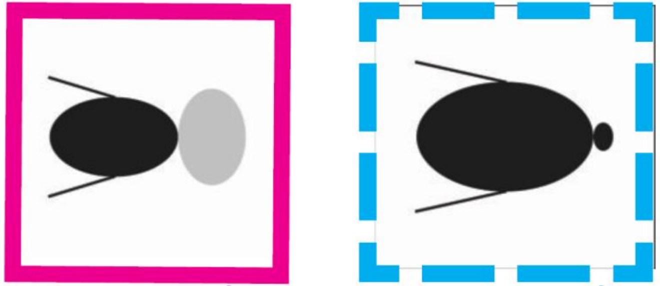
Crowdteaching Overview (1)

- Teaching set $\mathcal{X}$
  - (x, y) is a labeled example
  - $x \in \ \mathcal{X}$ and y $\in \{-1, 1\}$
- $\mathcal{H}$ is a finite set of hypotheses h
  - h: $\mathcal{X} \rightarrow \mathbb{R}$
- $\mathcal{X}$ is realizable
  - $h^* \in \mathcal{H}$

Crowdteaching Overview (2)

- Teach a worker $\mathcal{W}$ to classify $\mathcal{X}$ *well*
  - Select examples from $\mathcal{X}$
  - Steer $\mathcal{W}$ towards h^*
- $\mathcal{W}$ can *generalize* to new images

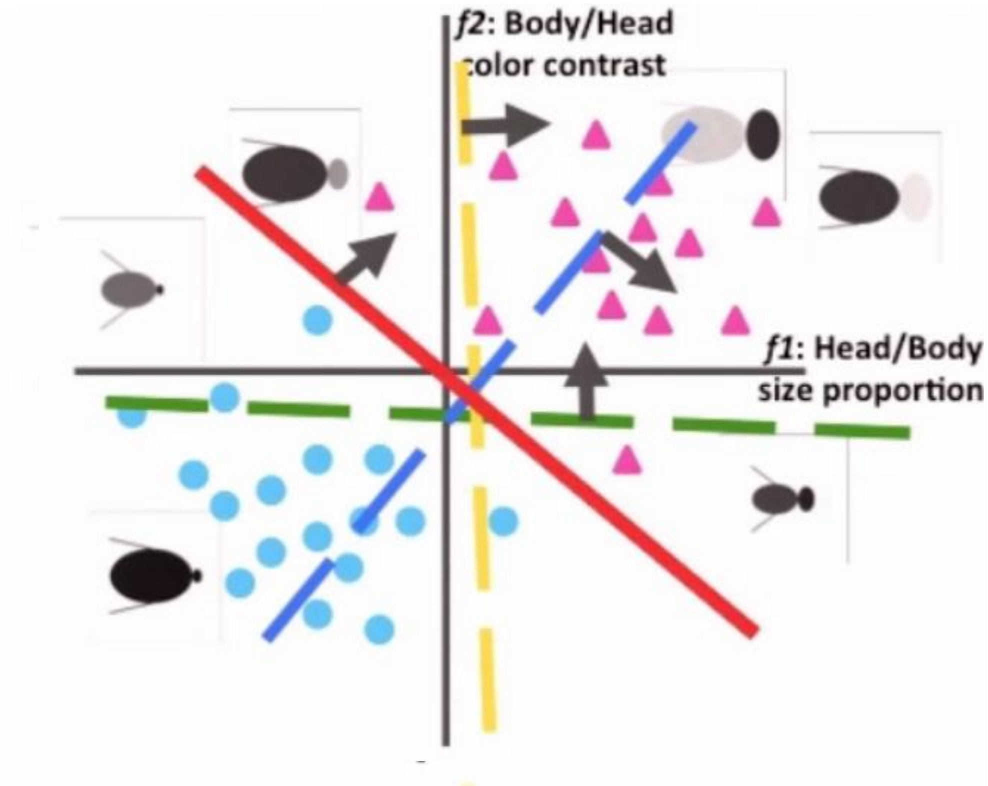
Example: $\mathcal{H}$ and h^*



Vespula Weevil

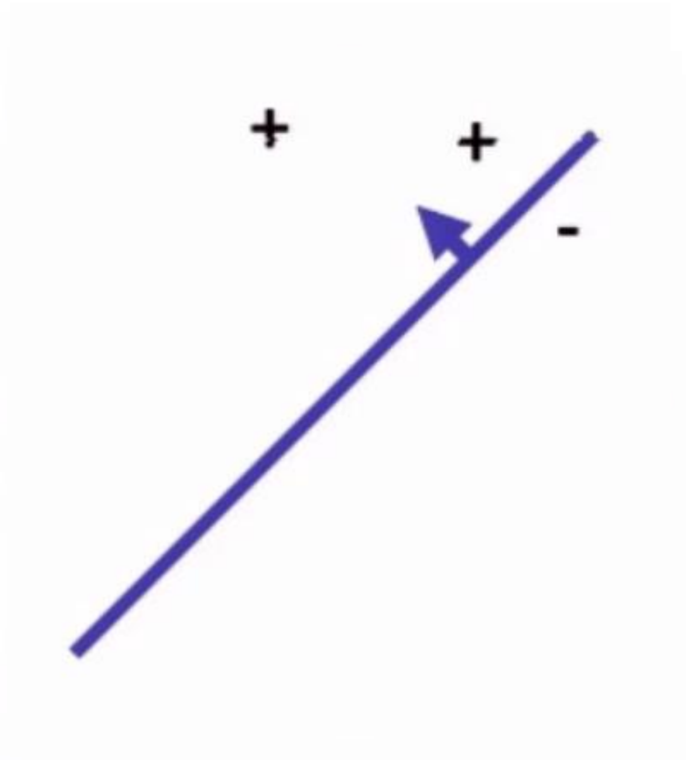
Hypothesis class $\mathcal{H}$

- Green: ignoring f1
- Yellow: ignoring f2
- Blue: wrongly using f2
- Red: correct hypothesis h^*



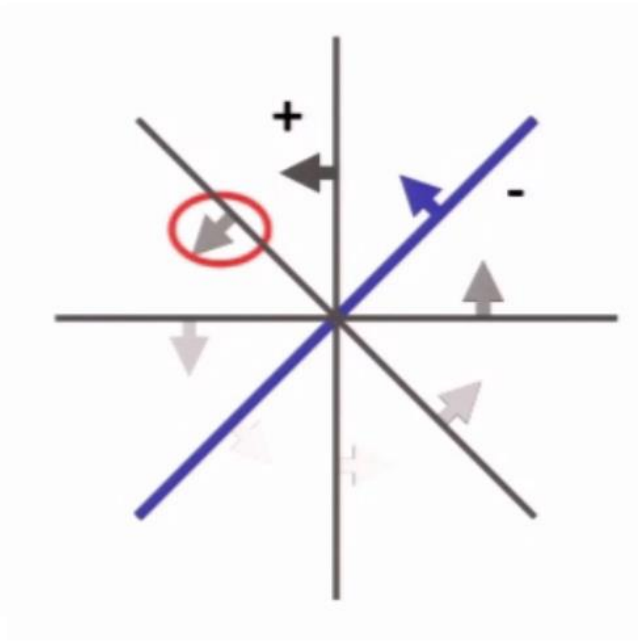
Noise-free Learners

- Learners immediately eliminate hypotheses inconsistent with observed examples
  - $h(x) \neq y$ for any (x, y) shown
- Tends to pick confusing order of examples

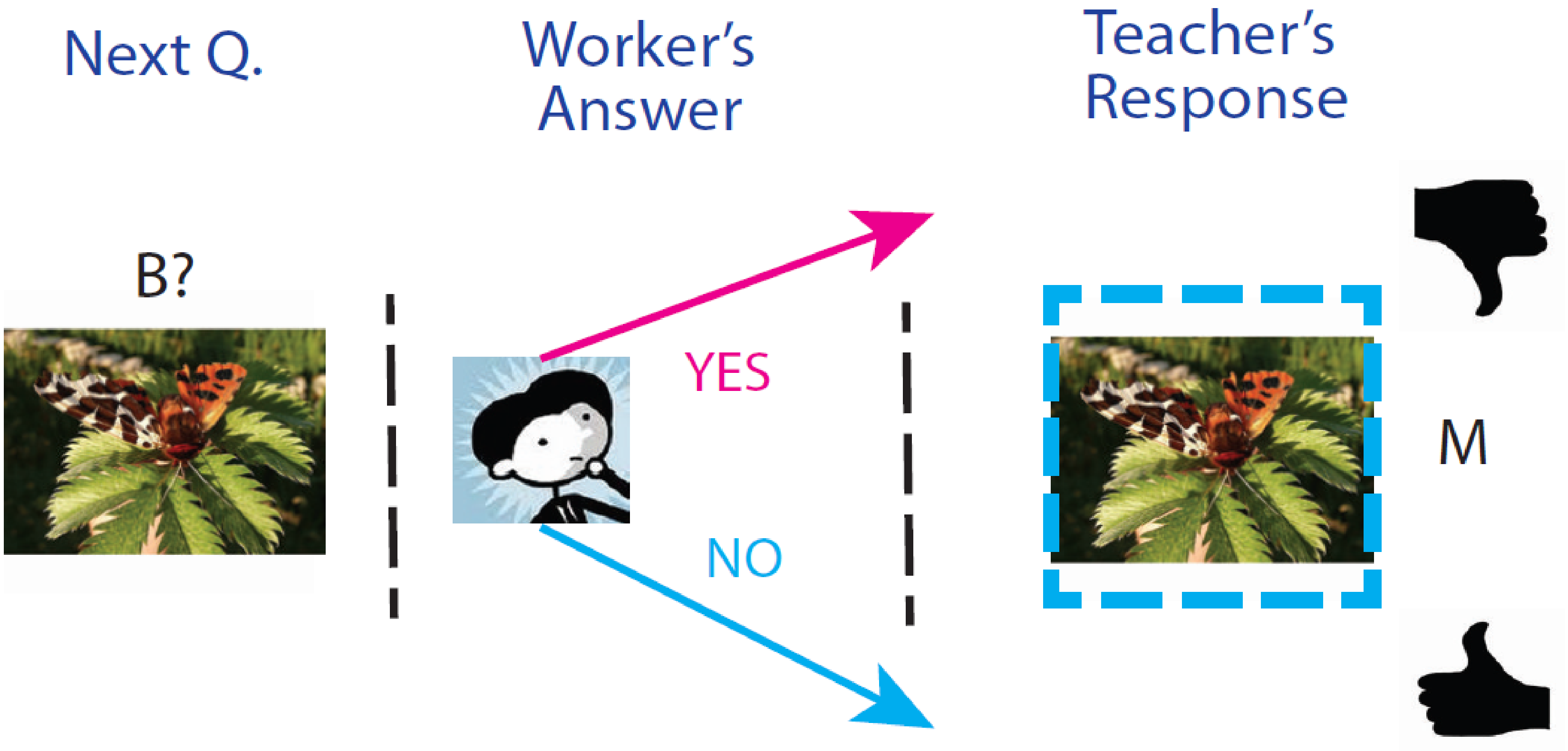


Noise-tolerant Learners

- Learner decreases the likelihood of hypotheses inconsistent with observed examples
  - Maintain weights for each hypothesis h
- On each example
  - Reduce weights for all inconsistent h
- Upon inconsistency
  - Jump to an h with a “large” weight
- Upon consistency
  - Stick with current h



Example: Teacher-Learner Interaction



The Teacher

- Find the smallest set of examples, $A \subseteq \mathcal{X}$, to achieve a desired learner error ϵ:

$$\boxed{A_\epsilon^* = \arg\min_{A \subseteq \mathcal{X}} |A| \;\; s.t. \;\; \mathbb{E}[err_L|A] \leq \epsilon}$$

- This optimization is NP-hard

STRICT Algorithm

- $F(A)$ evaluates the learner's expected error reduction after seeing A
- At each iteration:
  - Add the example that maximizes the error reduction

Policy 1 Teaching Policy STRICT

1: **Input:** examples $\mathcal{X}$, hyp. $\mathcal{H}$, prior P_0, error ϵ.
2: **Output:** teaching set A
3: $A \leftarrow \emptyset$
4: **while** $F(A) < \mathbb{E}[\mathrm{err}_L] - P_0(h^*)\epsilon$ **do**
5: $\quad x \leftarrow \arg\max_{x \in \mathcal{X}} (F(A \cup \{x\}))$
6: $\quad A \leftarrow A \cup \{x\}$
7: **end while**

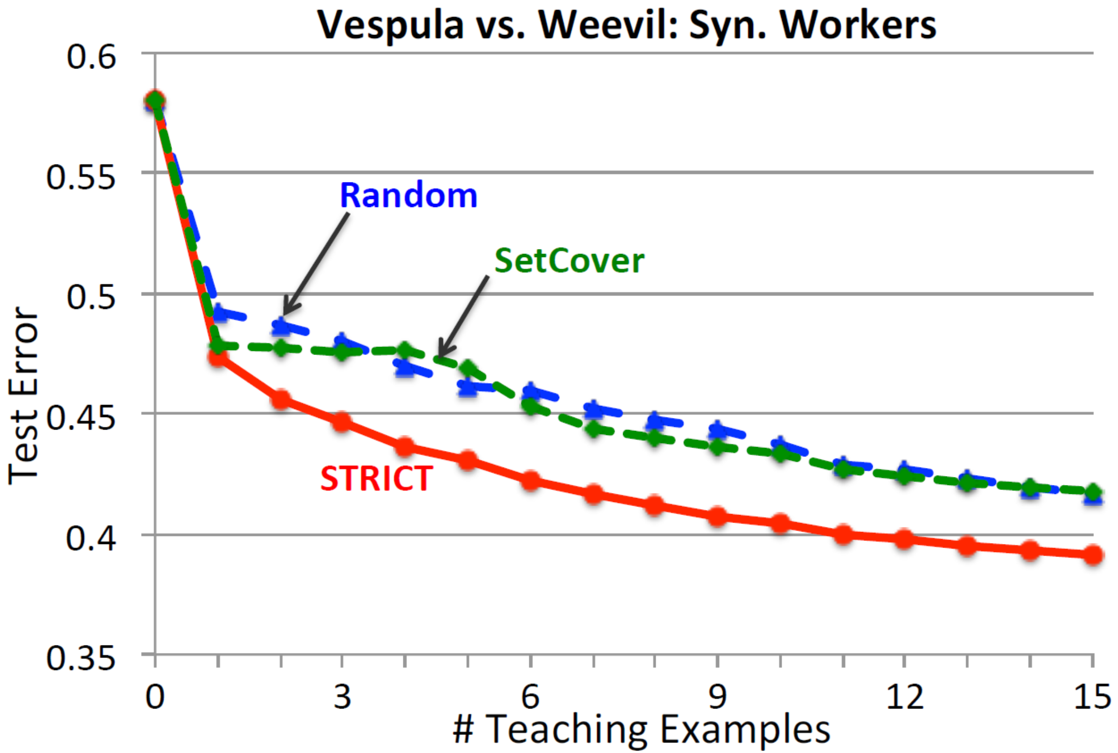
"Submodular Teaching for cRowdsourcIng ClassificaTion"

“Near-optimal”…?

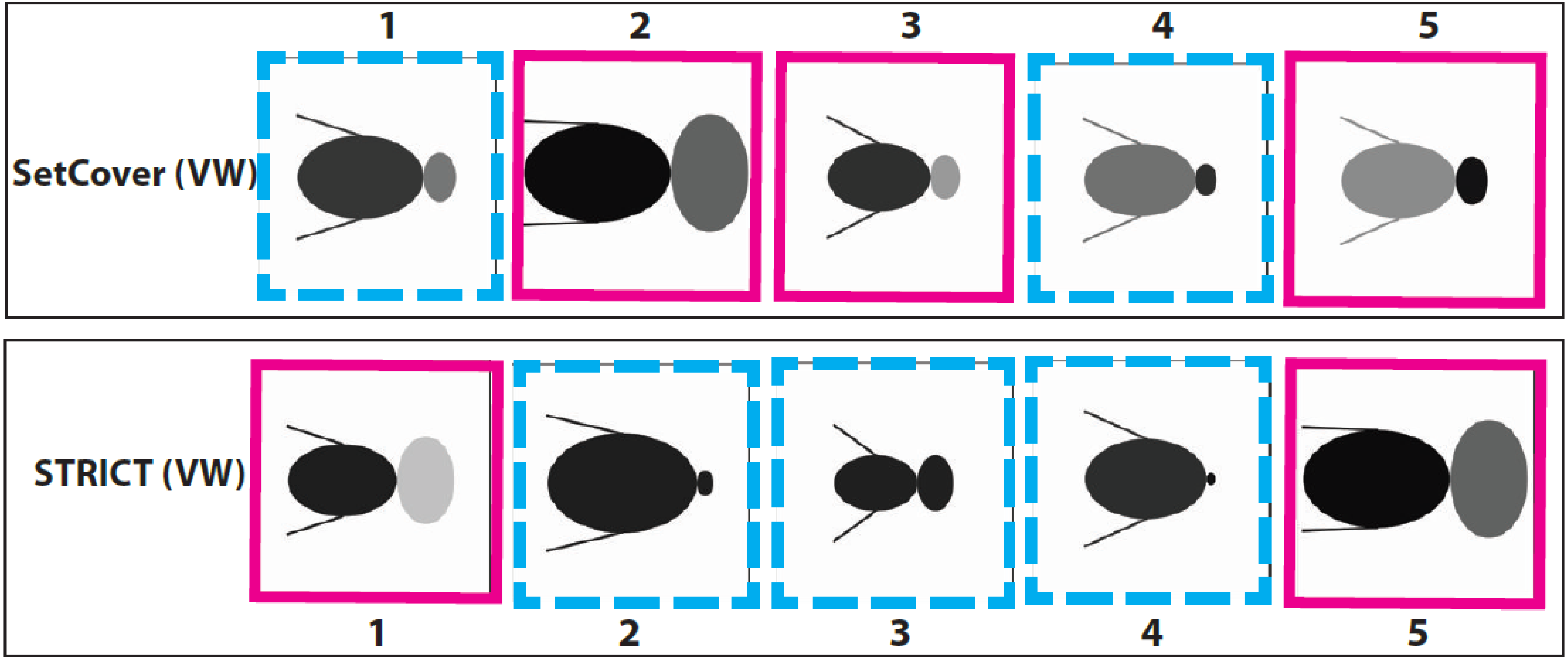
- $F(A)$ can be optimized using a greedy algorithm
- Theorem 1
  - Proof of approximation guarantees:
    - STRICT is not far from the optimal policy for ϵ

Experimental Results (Synthetic)

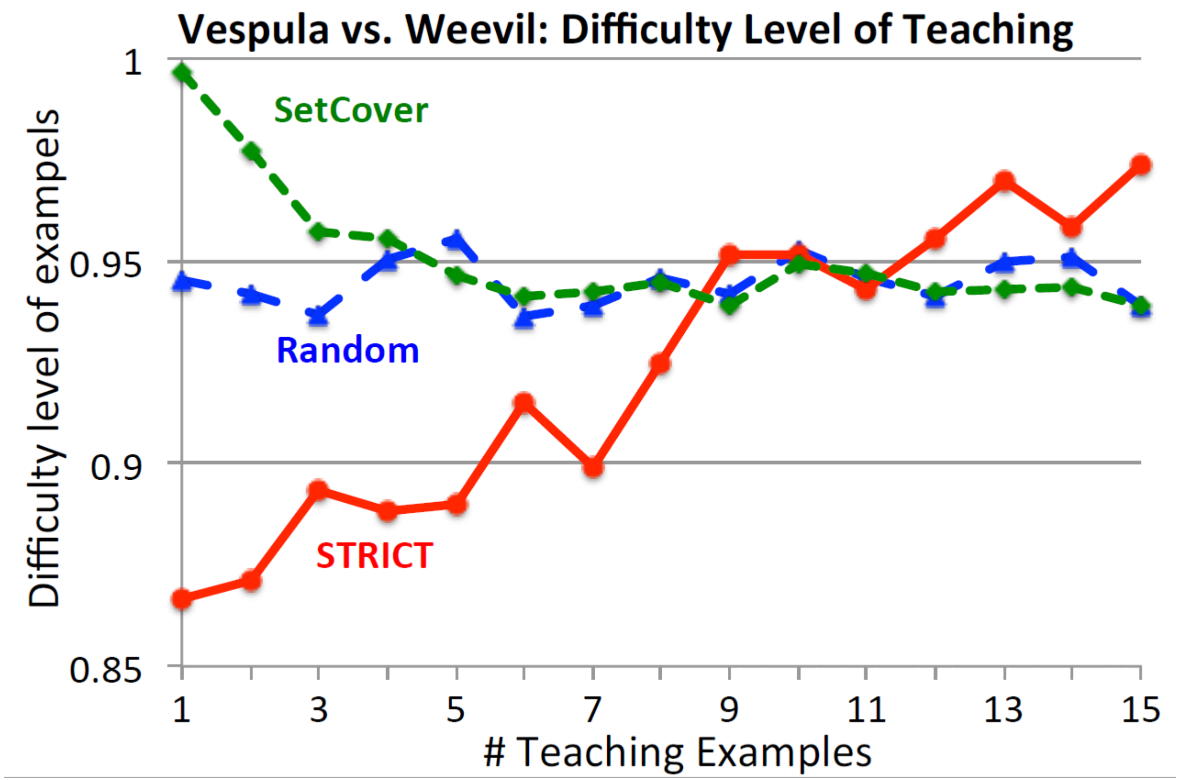
- Classify "Vespula" or "Weevil"
- 100 simulated learners
- Baselines:
  - Random
  - Noise-free learner



Teaching Sequence: VW



Difficulty of Selected Examples

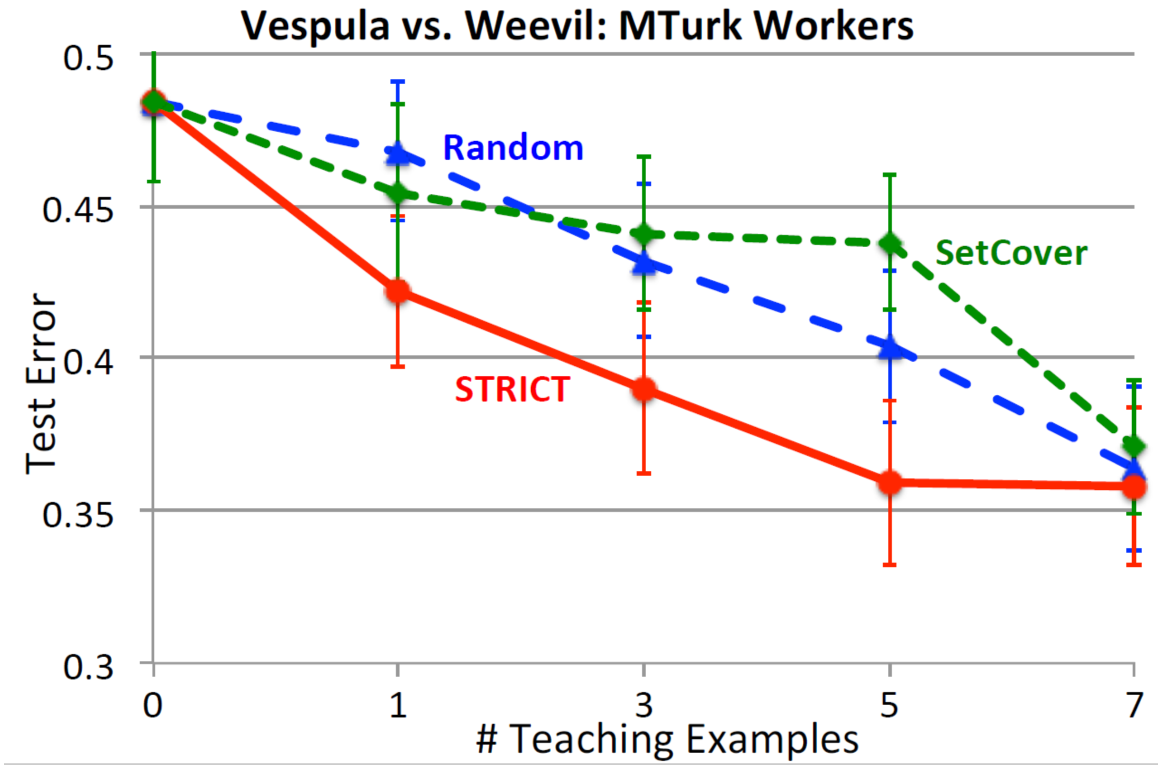


Experimental Results (MTurk)

- Different control groups
  - Length of teaching
  - Type of teaching
- Fixed payment for participation
  - Performance-based bonus

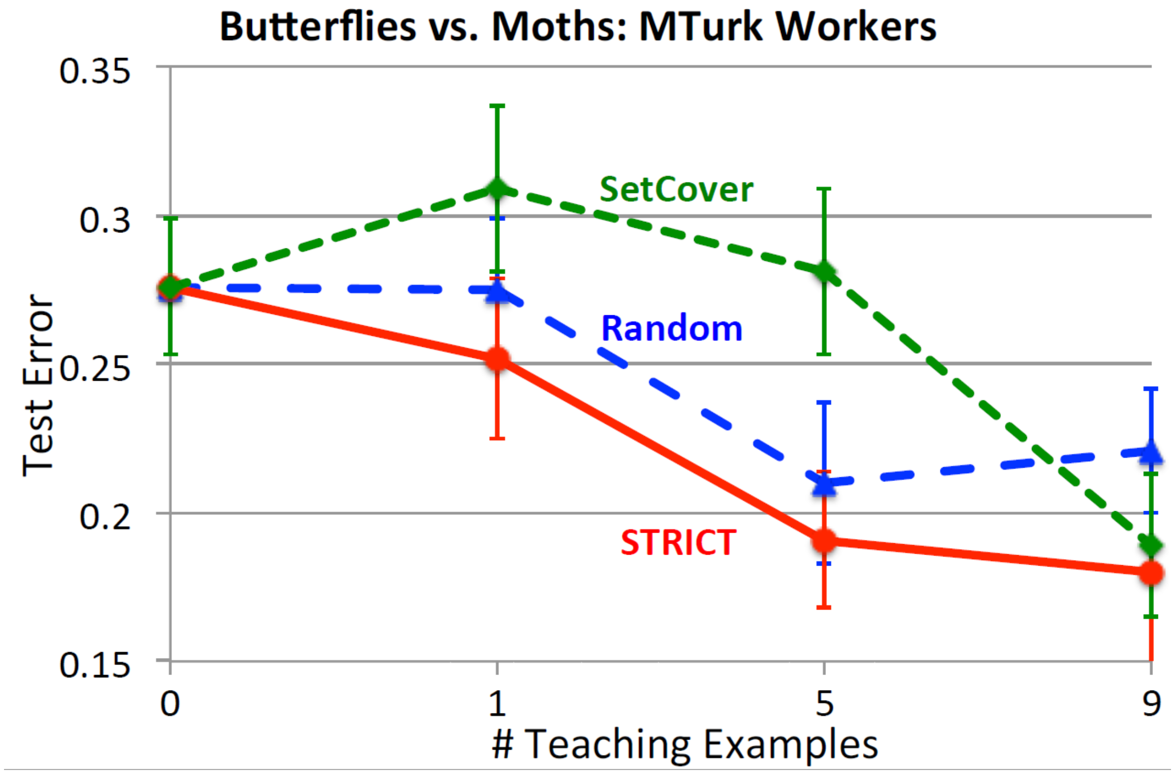
Vespula vs Weevil Results

- Classify Vespula or Weevil
- 780 annotators



Butterfly vs Moth Results

- Classify whether an images contains a butterfly or moth
- 300 annotators



STRICT Summary

- Noise-tolerant learners
- STRICT algorithm to teach workers
  - Guarantees near optimal examples are chosen
- Experimental results on MTurk
  - STRICT better than random and noise-free models

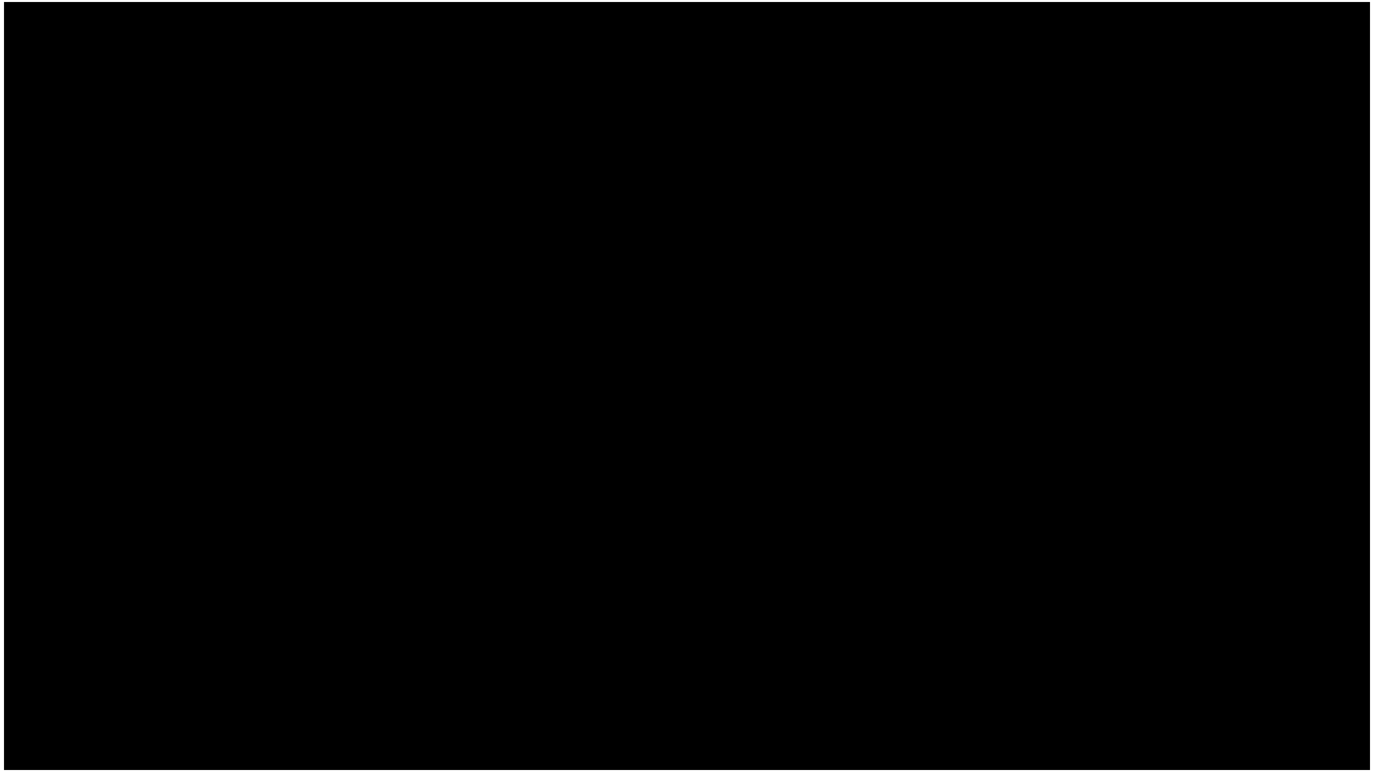
Related Work: Interactive Teaching

- Teacher obtains feedback from the learner
  - Which hypothesis are they using?
- Use feedback to select future teaching examples
- More difficult to employ in practice

Balbach and Zeugmann. Recent developments in algorithmic teaching. Proceedings of ICLATA, 2009. Zilles, Holte, and Zinkevich. Models of Cooperative Teaching and Learning. JMLR 2011. Du and Ling. Active Teaching for Inductive Learners, SIAM ICDM 2011., Doliwa, Dimon, and Zilles. Recursive Teaching Dimension, Learning Complexity, and Maximum Classes, Proceedings of ALT, 2010.

Related Work: Tutorials

- Interactive step by step tutorials
- Teaching a *skill*
- Photo editing using Photoshop



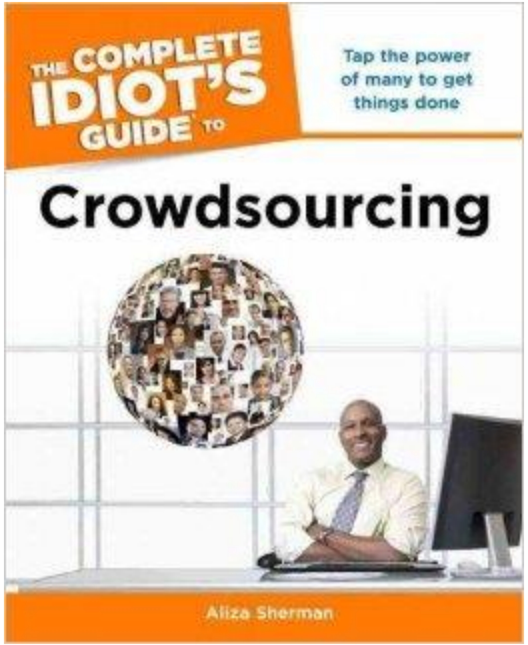
Mira Dontcheva, Robert R. Morris, Joel R. Brandt, and Elizabeth M. Gerber. 2014. Combining crowdsourcing and learning to improve engagement and performance. In *Proceedings of the 32nd annual ACM conference on Human factors in computing systems* (CHI '14). http://www.samasource.org/, https://www.mobileworks.com/

Discussion Qs

- Limitations of teaching with STRICT?
  - Are the STRICT input requirements realistic?
    - Feature set and ground truth labels ($\mathcal{X}$)
    - Hypothesis set ($\mathcal{H}$)
    - $\mathcal{X}$ is realizable ($h^* \in \mathcal{H}$)
- Alternative approaches to educating the crowd?
- Other applications of crowdteaching?

Related Resources

- http://las.ethz.ch/
- http://icml.cc/2014/
- http://crowdwisdom.cc/
- http://crowdresearch.org/blog/
- https://www.mturk.com/mturk/welcome
- http://techtalks.tv/beta/talks/near-optimally-teaching-the-crowd-to-classify/61069/
- http://techtalks.tv/talks/submodularity-in-machine-learning-new-directions-part-i/58125/



$7.98 on Amazon

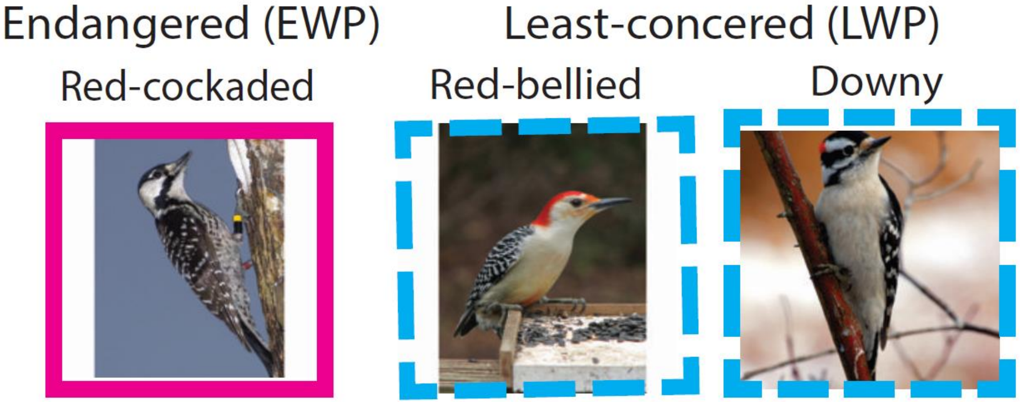
Questions?

http://www.crowdsourcing.org/editorial/growing-your-business-with-crowdsourcing/23054

BACKUP

Classifying Endangered Species

- Classify if an image contains a Red-cockaded Woodpecker or not
- Red-cockaded → +
- Red-bellied or Downy → -
- Feature space
  - Metadata from the crowd
  - 312 binary attributes



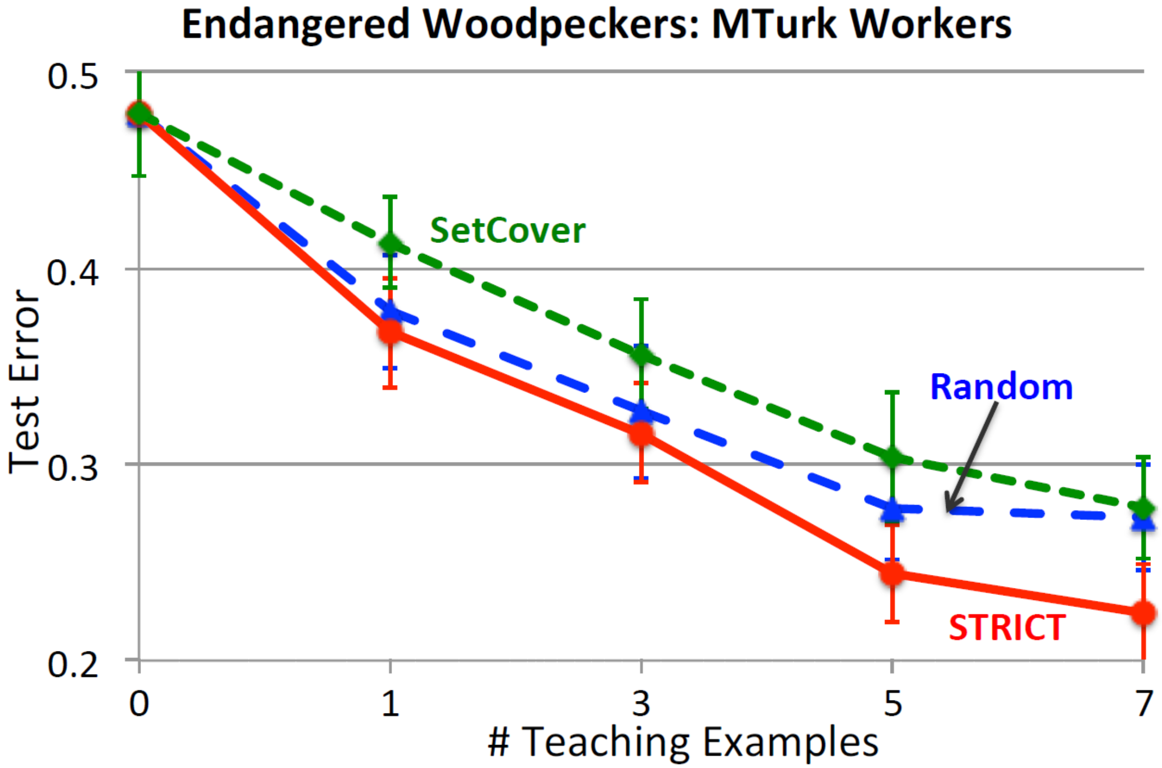
Brian L. Sullivan, Christopher L. Wood, Marshall J. Iliff, Rick E. Bonney, Daniel Fink, Steve Kelling. eBird: A citizen-based bird observation network in the biological sciences.

Woodpecker Features

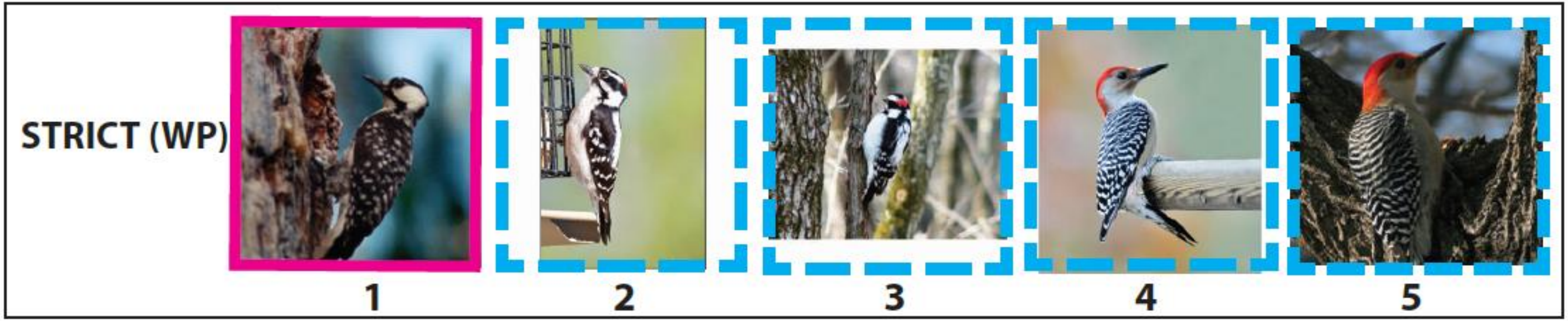
Features	% feature presence in species			h*
	Red-cockaded	Red-bellied	Downy	
has_breast_pattern: *solid*	22	88	90	-1
has_upper_tail_color: *white*	55	91	53	0
has_bill_length: *same_as_head*	78	93	25	0
has_bill_length: *shorter_than_head*	23	7	76	0
has_forehead_color: *black*	91	0	96	0
has_forehead_color: *red*	0	78	0	0
has_nape_color: *black*	85	0	93	0
has_nape_color: *red*	0	87	15	0
has_back_pattern: *spotted*	75	27	29	0
has_back_pattern: *striped*	18	69	7	0
has_belly_pattern: *solid*	25	81	94	-1
has_crown_color: *black*	98	0	96	0
has_crown_color: *red*	2	86	36	-1

Endangered Woodpeckers Results

- Classifying Endangered or Not endangered
- Crowd assignment of features
- 13 binary attributes
- 520 annotators



Teaching Sequence: EN

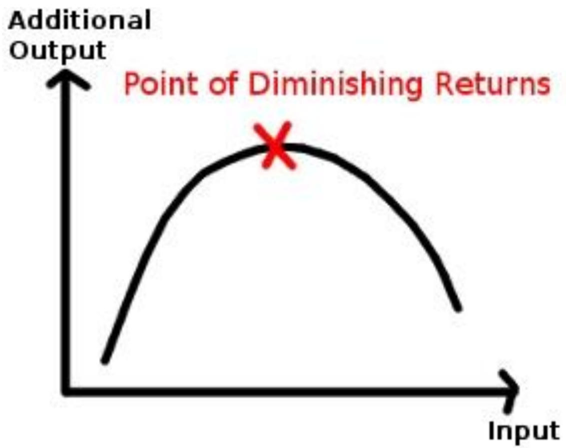


Set Optimization Problems

- Problem: Optimize a set function F that scores utility (or cost) on a set A
- If F is submodular, we can:
  - Solve optimization problems with strong guarantees
- Marginal gain
  - How much does my utility increase as I add another item to this set?

Submodular Functions

- Submodular functions have natural diminishing returns
  - Add an item to a “small” set → bigger gain
  - Add an item to a “large” set → smaller gain
- Marginal gain of adding an item to a set eventually gets smaller as the set gets bigger



Example: Diminishing Return

- Determining the optimal number of ceiling sensors in a smarthome
- Each sensor has an *area* it covers
- The utility of a set A of sensors is measured by the area collectively covered by A
- The marginal gain of adding sensors to A eventually maxes out
  - Gain then begins to decrease

Teaching as Set Optimization

- A is a set of examples
- $F(A)$ scores the expected error reduction of the learner after seeing set A
- Maximizing F yields the set with the largest error reduction

STRICT Objective Function

Uncertainty about h *before* seeing A

Uncertainty about h *after* seeing A

Fraction of all examples on which h is wrong

$$F(A) = \sum_{h \in \mathcal{H}} \left(Q(h) - Q(h|A) \right) \mathrm{err}(h, h^*) \text{ , where}$$

$$Q(h|A) = P_0(h) \prod_{\substack{x \in A \\ y(x) \neq \mathrm{sgn}(h(x))}} P(y(x)|h, x)$$

Thank you!