Active Learning by Learning

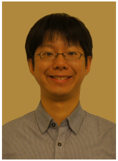
Hsuan-Tien Lin (林軒田)
htlin@csie.ntu.edu.tw

Department of Computer Science
& Information Engineering

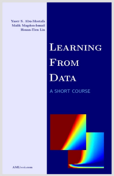
National Taiwan University
(國立台灣大學資訊工程系)

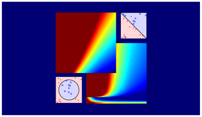
2015 IR Workshop, IIS Sinica, Taiwan
joint work with Wei-Ning Hsu, presented in AAAI 2015

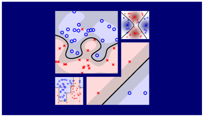
About Me
Hsuan-Tien Lin



- Associate Professor, Dept. of CSIE, National Taiwan University
- Leader of the Computational Learning Laboratory
- Co-author of the textbook “*Learning from Data: A Short Course*” (often **ML best seller on Amazon**)
- Instructor of the NTU-Coursera Mandarin-teaching ML Massive Open Online Courses
  - “*Machine Learning Foundations*”: `www.coursera.org/course/ntumlone`
  - “*Machine Learning Techniques*”: `www.coursera.org/course/ntumltwo`







Apple Recognition Problem

Note: Slide Taken from my “ML Techniques” MOOC

- need **apple classifier**: is this a picture of an apple?
- gather photos under CC-BY-2.0 license on Flicker (**thanks to the authors below!**) and **label them as apple/other for learning**

(APAL stands for Apple and Pear Australia Ltd)


Dan Foy
`https://flic.kr/p/jNQ55`


APAL
`https://flic.kr/p/jzP1VB`


adrianbartel
`https://flic.kr/p/bdy2hZ`


ANdrzej cH.
`https://flic.kr/p/51DKA8`


Stuart Webster
`https://flic.kr/p/9C3Ybd`


nachans
`https://flic.kr/p/9XD7Ag`


APAL
`https://flic.kr/p/jzRe4u`


Jo Jakeman
`https://flic.kr/p/7jwtGp`


APAL
`https://flic.kr/p/jzPYNr`


APAL
`https://flic.kr/p/jzScif`

Apple Recognition Problem

Note: Slide Taken from my “ML Techniques” MOOC

- need **apple classifier**: is this a picture of an apple?
- gather photos under CC-BY-2.0 license on Flicker (**thanks to the authors below!**) and **label them as apple/other for learning**

Mr. Roboto.

https://flic.kr/p/i5BN85

Richard North

https://flic.kr/p/bHhPkB

Richard North

https://flic.kr/p/d8tGou

Emilian Robert Vicol

https://flic.kr/p/bpmGXW

Nathaniel McQueen

https://flic.kr/p/pZv1Mf

Crystal

https://flic.kr/p/kaPYp

jfh686

https://flic.kr/p/6vjRFH

skyseeker

https://flic.kr/p/2MynV

Janet Hudson

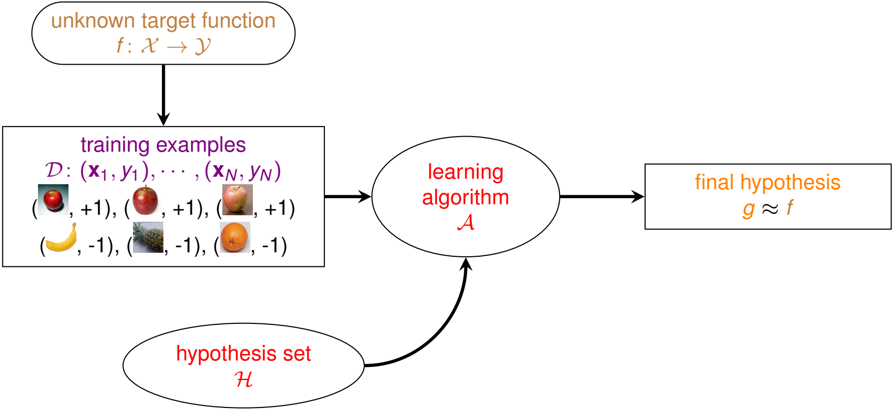
https://flic.kr/p/7QDBbm

Rennett Stowe

https://flic.kr/p/agmnrk

Batch (Traditional) Machine Learning

Note: Flow Taken from my “ML Foundations” MOOC



unknown target function
$f: \mathcal{X} \to \mathcal{Y}$

training examples
$\mathcal{D}: (\mathbf{x}_1, y_1), \cdots, (\mathbf{x}_N, y_N)$
(, +1), (, +1), (, +1)
(, -1), (, -1), (, -1)

learning
algorithm
$\mathcal{A}$

final hypothesis
$g \approx f$

hypothesis set
$\mathcal{H}$

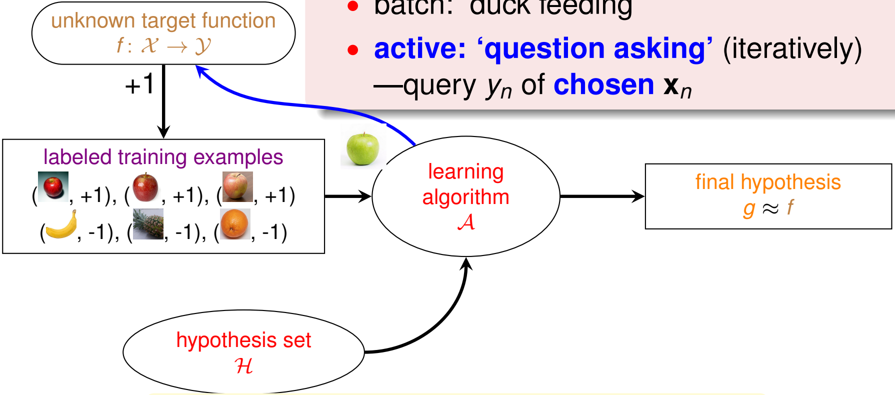
batch supervised classification:
learn from **fully labeled** data

Active Learning: Learning by 'Asking'

but labeling is **expensive**

Protocol ⇔ Learning Philosophy

- batch: 'duck feeding'
- **active: 'question asking'** (iteratively) —query y_n of **chosen** $\mathbf{x}_n$



unknown target function
$f \colon \mathcal{X} \to \mathcal{Y}$

+1

labeled training examples

(, +1), (, +1), (, +1)

(, -1), (, -1), (, -1)

learning algorithm
$\mathcal{A}$

final hypothesis
$g \approx f$

hypothesis set
$\mathcal{H}$

active: improve hypothesis with fewer labels (hopefully) by asking questions **strategically**

Pool-Based Active Learning Problem

Given

- labeled pool $\mathcal{D}_l = \left\{(\text{feature } \mathbf{x}_n$  , **label** y_n (e.g. IsApple?)$)\right\}_{n=1}^{N}$
- unlabeled pool $\mathcal{D}_u = \left\{\tilde{\mathbf{x}}_s\right\}_{s=1}^{S}$

Goal

design an algorithm that iteratively

1. **strategically query** some $\tilde{\mathbf{x}}_s$  to get associated $\tilde{y}_s$
2. move $(\tilde{\mathbf{x}}_s, \tilde{y}_s)$ from $\mathcal{D}_u$ to $\mathcal{D}_l$
3. learn **classifier** $g^{(t)}$ from $\mathcal{D}_l$

and improve **test accuracy of** $g^{(t)}$ w.r.t **#queries**

how to **query strategically**?

How to Query Strategically?

by DFID - UK Department for International Development;

licensed under CC BY-SA 2.0 via Wikimedia Commons

Strategy 1

ask **most confused** question

Strategy 2

ask **most frequent** question

Strategy 3

ask **most helpful** question

do you use a **fixed strategy** in practice? ☺

Choice of Strategy

Strategy 1:
uncertainty

ask **most confused** question

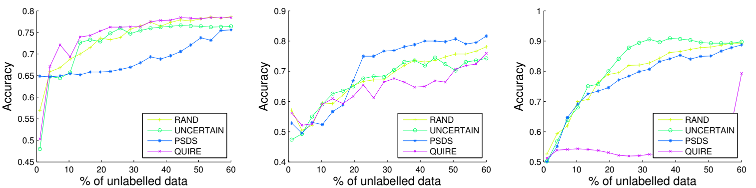
Strategy 2:
representative

ask **most frequent** question

Strategy 3:
exp.-err. reduction

ask **most helpful** question

- **choosing** one single strategy is **non-trivial**:



- human-designed strategy **heuristic** and **confine** machine's ability

can we **free** the machine ☺
by letting it **learn to choose** the strategies?

Our Contributions

a philosophical and algorithmic study of active learning, which ...

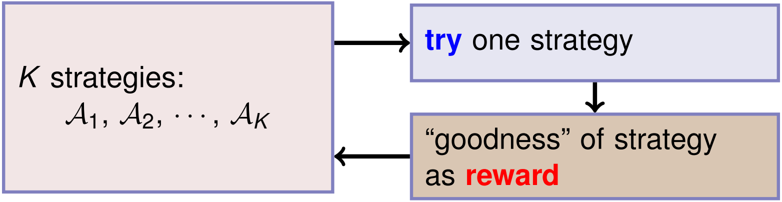
- allows machine to make **intelligent choice of strategies**, just like my cute daughter
- studies **sound feedback scheme** to tell machine about goodness of choice, just like what I do
- results in **promising active learning performance**, just like (hopefully) bright future of my daughter ☺

will describe **key philosophical ideas**
behind our proposed approach

Idea: Trial-and-Reward Like Human

by DFID - UK Department for International Development;

licensed under CC BY-SA 2.0 via Wikimedia Commons



K strategies:
$\mathcal{A}_1, \mathcal{A}_2, \cdots, \mathcal{A}_K$

try one strategy

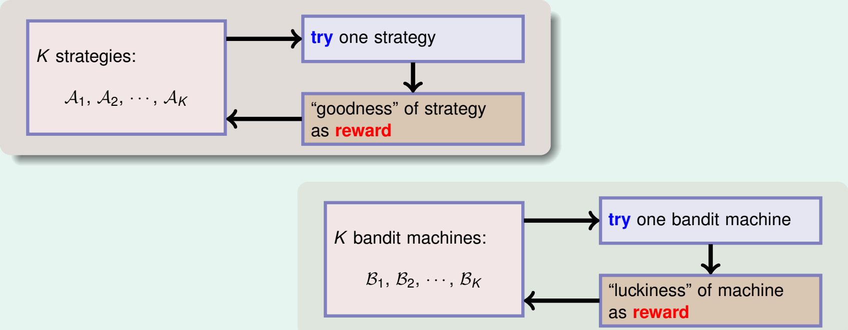
"goodness" of strategy as **reward**

two issues: **try** and **reward**

Reduction to Bandit

when do humans **trial**-and-**reward**?

gambling ☺



K strategies:

$\mathcal{A}_1, \mathcal{A}_2, \cdots, \mathcal{A}_K$

try one strategy

"goodness" of strategy as **reward**

K bandit machines:

$\mathcal{B}_1, \mathcal{B}_2, \cdots, \mathcal{B}_K$

try one bandit machine

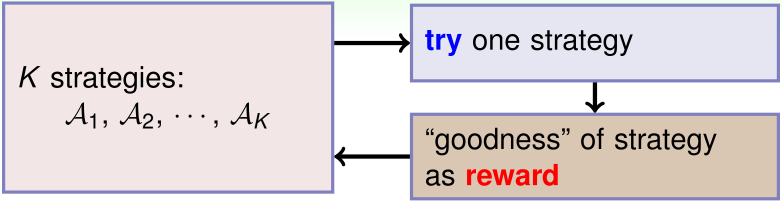
"luckiness" of machine as **reward**

—will take one well-known **probabilistic bandit learner (EXP4.P)**

intelligent choice of strategy

$\Longrightarrow$ intelligent choice of **bandit machine**

Active Learning by Learning



K strategies:
$\mathcal{A}_1, \mathcal{A}_2, \cdots, \mathcal{A}_K$

try one strategy

"goodness" of strategy as **reward**

Given: K existing active learning strategies

for $t = 1, 2, \ldots, T$

1. let EXP4.P **decide strategy $\mathcal{A}_k$ to try**
2. **query the $\tilde{x}_s$** suggested by $\mathcal{A}_k$, and compute $g^{(t)}$
3. evaluate **goodness of $g^{(t)}$** as **reward** of **trial** to update EXP4.P

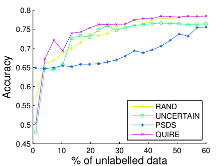
only remaining problem: **what reward**?

Ideal Reward

ideal reward after updating classifier $g^{(t)}$ by the query $(\mathbf{x}_{n_t}, y_{n_t})$:

$$\text{accuracy } \frac{1}{M}\sum_{m=1}^{M} \llbracket y_m = g^{(t)}(\mathbf{x}_m) \rrbracket \text{ on } \textbf{test set } \{(\mathbf{x}_m, y_m)\}_{m=1}^{M}$$

- **test accuracy** as reward:
  area under query-accuracy curve ≡ cumulative reward



- test accuracy **infeasible** in practice
  —labeling **expensive**, remember? ☺

difficulty: approximate test accuracy **on the fly**

Training Accuracy as Reward

test accuracy $\xcancel{\frac{1}{M}\sum_{m=1}^{M}[\![y_m = g^{(t)}(\mathbf{x}_m)]\!]}$ infeasible, naïve replacement:

accuracy $\frac{1}{t}\sum_{\tau=1}^{t}\left[\!\!\left[y_{n_\tau} = g^{(t)}(\mathbf{x}_{n_\tau})\right]\!\!\right]$ on **labeled pool** $\{(\mathbf{x}_{n_\tau}, y_{n_\tau})\}_{\tau=1}^{t}$

- **training accuracy** as reward:
  training accuracy $\approx$ **test accuracy**?
- not necessarily!!
  —for active learning strategy that asks **easiest** questions:
  - training accuracy **high**: $\mathbf{x}_{n_\tau}$ easy to label
  - test accuracy **low**: not enough information about **harder instances**

training accuracy:
too **biased** to approximate test accuracy

Weighted Training Accuracy as Reward

training accuracy ~~$\frac{1}{t}\sum_{\tau=1}^{t}[\![y_{n_\tau} = g^{(t)}(\mathbf{x}_{n_\tau})]\!]$~~ biased,
want **unbiased estimator**

- **non-uniform sampling** theorem: if $(\mathbf{x}_{n_\tau}, y_{n_\tau})$ sampled with probability $p_\tau > 0$ from data set $\{(\mathbf{x}_n, y_n)\}_{n=1}^{N}$ in iteration τ,

  weighted training accuracy $$\frac{1}{t}\sum_{\tau=1}^{t}\frac{1}{p_\tau}[\![y_{n_\tau} = g(\mathbf{x}_{n_\tau})]\!]$$

  $$\approx \quad \frac{1}{N}\sum_{n=1}^{N}[\![y_n = g(\mathbf{x}_n)]\!]$$ in **expectation**

- with **probabilistic query** like EXP4.P:
  weighted training accuracy $\approx$ test accuracy

weighted training accuracy:
unbiased approx. of test accuracy on the fly

Human-Designed Criterion as Reward

(Baram et al., 2004) COMB approach:

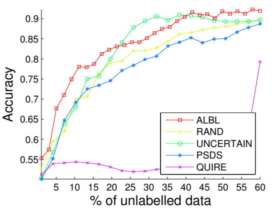
bandit + **balancedness** of $g^{(t)}$ on unlabeled data as reward

- why? human criterion that matches classifier to **domain assumption**
- but many active learning applications are on **unbalanced data!** —assumption may be **unrealistic**

existing strategies: active learning by acting;
COMB: active learning by acting;
ours: active learning **by learning**

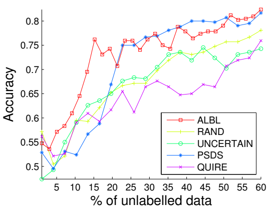
Comparison with Single Strategies

UNCERTAIN Best



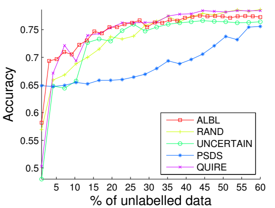
`vehicle`

PSDS Best



`sonar`

QUIRE Best



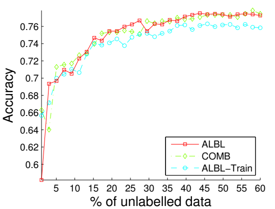
`diabetes`

- **no single best strategy** for every data set
  —choosing/blending needed
- **ALBL** consistently **matches the best**
  —similar findings across other data sets

ALBL: effective in **making intelligent choices**

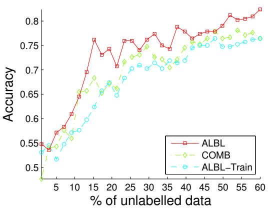
Comparison with Other Adaptive Blending Algorithms

ALBL $\approx$ COMB



`diabetes`

ALBL $>$ COMB



`sonar`

- **ALBL** $>$ ALBL-Train generally
  —**importance-weighted** mechanism needed for
  correcting biased training accuracy
- **ALBL** consistently **comparable to or better than** COMB
  —**learning performance** more useful than human-criterion

ALBL: effective in **utilizing performance**

Conclusion

Active Learning by Learning

- based on **bandit learning** + **unbiased performance estimator** as reward
- effective in **making intelligent choices**
  —comparable or superior to the best of existing strategies
- effective in **utilizing learning performance**
  —superior to human-criterion-based blending

New Directions

- **open-source tool** being developed
- extending to **more sophisticated active learning problems**

Thank you! Questions?