Making Active Learning More Realistic

Hsuan-Tien Lin

National Taiwan University

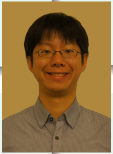
htlin@csie.ntu.edu.tw

Keynote Talk of Weakly-supervised Learning Workshop @ ACML
November 17 2019

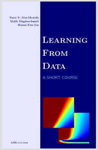
About Me

Professor
National Taiwan University

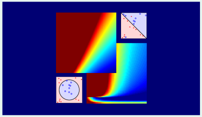
Chief Data Science Consultant
(former Chief Data Scientist)
Appier Inc.

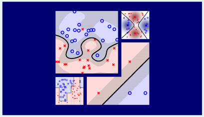


Co-author
Learning from Data



Instructor
NTU-Coursera MOOCs
ML Foundations/Techniques





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 Mc-Queen

`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

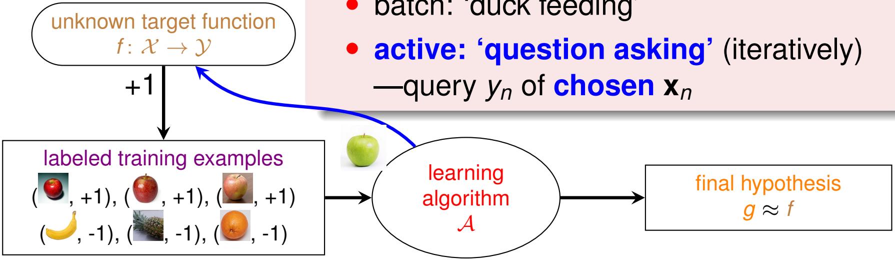
`https://flic.kr/p/agmnrk`

Active Learning: Learning by 'Asking'

but labeling is **expensive**

Protocol ⇔ Learning Philosophy

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



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

Pool-Based Active Learning Problem

Given

- labeled pool $\mathcal{D}_l = \left\{ (\text{feature } \mathbf{x}_n \right.$  $\left., \textbf{label } y_n \text{ (e.g. IsApple?)}) \right\}_{n=1}^{N}$
- 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 $\mathcal{D}_l$ (optionally, $+\mathcal{D}_u$)

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

how to **query strategically**?

How to Query Strategically? (1/2)

by DFID - UK Department for International Development;
licensed under CC BY-SA 2.0 via Wikimedia Commons

Strategy 1

ask **most confused** question

how to define **confusion (uncertainty)**? ☺

Uncertainty of Hard Binary Classifier

uncertain $\Leftrightarrow$ **near binary decision boundary**

Active Learning with SVM (Tong, 2000)

1. learn a SVM hyperplane with $\mathcal{D}_l$ (blue squares)
2. query $\tilde{\mathbf{x}}_s$ (magenta circle) closest to the hyperplane

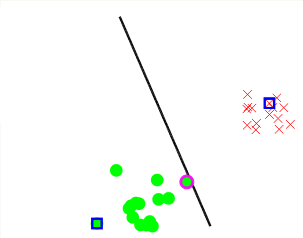
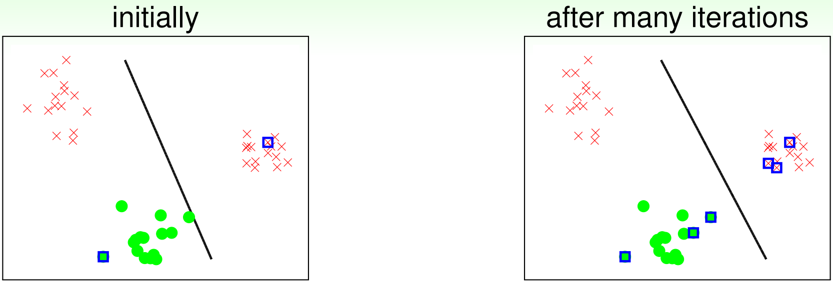


figure from my former student Chun-Liang Li's ACML 2012 presentation (Li, 2012)

uncertainty sampling:
arguably **most popular** AL paradigm
because **simple** and **effective**

Problem of Uncertainty Sampling



figures from my student Chun-Liang Li's ACML 2012 presentation (Li, 2012)

- **overly confident** about unknown clusters
- trapped in bad **"local optimal"**

solution: query **unknown clusters**

How to Query Strategically? (2/2)

Strategy 2

ask **most frequent** question

how to **combine frequency (density) with uncertainty**? ☺

Representative Sampling (1/2): Query of Informative and Repre. Examples (QUIRE)

QUIRE (Huang, 2010)

$$\tilde{\mathbf{x}}_s = \underset{\tilde{\mathbf{x}}}{\operatorname{argmin}} \left(\text{worst case loss on } (\mathcal{D}_l \cup \mathcal{D}_u) \mid (\tilde{\mathbf{x}}, \tilde{y}) \right)$$

where loss $\approx$ -accuracy

- $\approx$ uncertainty: knowing $\tilde{y}$ improves loss a lot **in worst case**
- $\approx$ representative: knowing $\tilde{y}$ improves loss a lot **for $\mathcal{D}_u$ part**

QUIRE: promising for RS,
but **time consuming** to compute

Representative Sampling (2/2): Hinted Sampling

Uncertainty Sampling (US)

$$\tilde{\mathbf{x}}_s = \operatorname*{argmax}_{\tilde{\mathbf{x}}} \left(1 - |p(\tilde{\mathbf{x}})|\right)$$

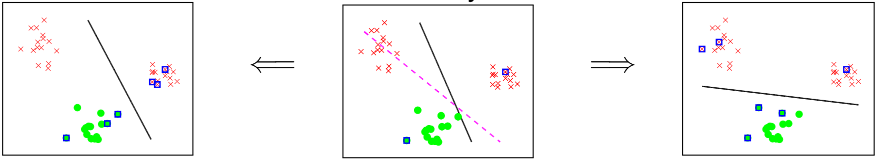
where $p(\tilde{\mathbf{x}})$ is probability of more likely class

Hinted Sampling

$$\tilde{\mathbf{x}}_s = \operatorname*{argmax}_{\tilde{\mathbf{x}}} \left(1 - |p'(\tilde{\mathbf{x}})|\right)$$

where $p'(\tilde{\mathbf{x}}) \approx \frac{1}{2}$ for **dense regions of $\mathcal{D}_u$**

afterwards $\Longleftarrow$ initially $\Longrightarrow$ afterwards



figures from my student Chun-Liang Li's ACML 2012 presentation (Li, 2012)

Hinted Sampling: **simple realization** of RS

How to Query Strategically?

by DFID - UK Department for International Development;

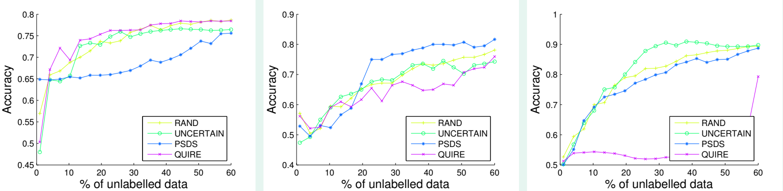
Strategy 1	Strategy 2	Strategy 3
uncertainty sampling	QUIRE	hinted sampling

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

Choice of Strategy

Strategy 1	Strategy 2	Strategy 3
uncertainty sampling	QUIRE	~~hinted sampling~~ PSDS (Donmez , 2008)

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



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

choice (in advance): practitioner's pain point
—proposed solution: **learning to choose**

Idea: Trial-and-Reward Like Human

by DFID - UK Department for International Development;

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 (Hsu, 2015)

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

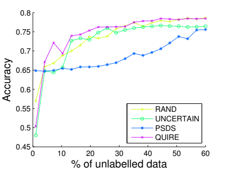
only remaining problem: **what reward**?

Ideal Reward

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

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

- **test accuracy** as reward:
  area under query-accuracy curve $\equiv$ cumulative reward



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

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

Weighted Training Accuracy as Reward

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

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

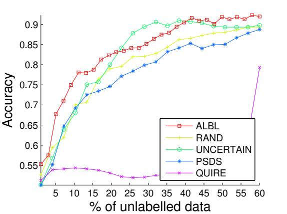
$$\approx \quad \frac{1}{N}\sum_{n=1}^{N} [\![y_n = g(\mathbf{x}_n)]\!] \text{ in } \textbf{expectation}$$

- with **probabilistic query** like EXP4.P:
  weighted training accuracy $\approx$ test accuracy
  —can approximate test accuracy on the fly

ALBL: EXP4.P + **weighted** training accuracy

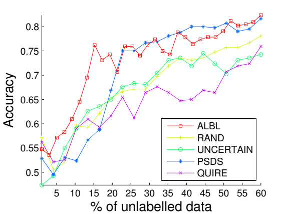
Comparison with Single Strategies

UNCERTAIN Best



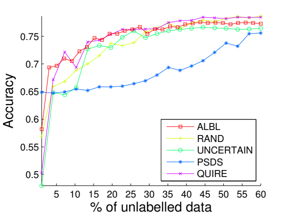
vehicle

PSDS Best



sonar

QUIRE Best

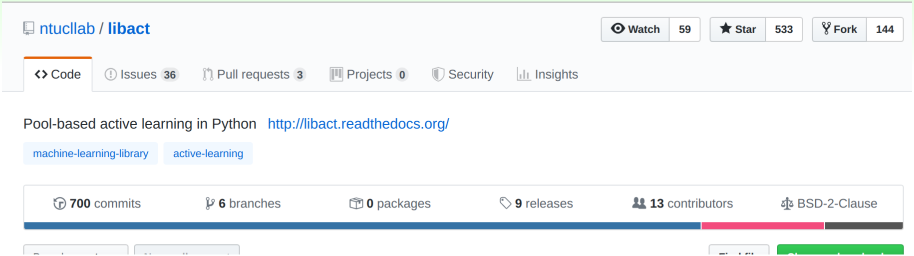


diabetes

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

ALBL: effective in **making intelligent choices**

Have We Made Active Learning More Realistic? (1/2)



Yes!

open-source tool `libact` developed (Yang, 2017)

`https://github.com/ntucllab/libact`

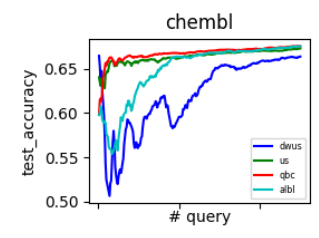
- including uncertainty, hinted, QUIRE, PSDS **and ALBL**
- received $>$ 500 **stars** and continuous **issues**

"`libact` is a Python package designed to **make active learning easier** for real-world users"

Have We Made Active Learning More Realistic? (2/2)

No!

- single-most raised **issue**: hard to install on Windows/Mac —because **hinted** (& others) requires some C packages
- performance in a recent industry project:



  - **uncertainty** sampling **often suffices**
  - ALBL **dragged down by bad strategy** (DWUS: representative sampling)

> "`libact` is a Python package **designed to** make active learning easier for real-world users"

Other Attempts for Realistic Active Learning

"learn" a strategy **beforehand** rather than on-the-fly?

- learning active learning (Konyushkova, 2017)
- transfer active learning experience (Chu, 2016)

—**not easy to realize** in open-source package

strategy to save **true resource consumption (cost)**?

- annotation cost-sensitive learning (Tsou, 2019)

—**costly to get costs** ☺

many more needs to be satisfied: mini-batch, multi-label query, **weak-label query**, etc.

Summary

Traditional Active Learning

- strategies by **human philosophy**: uncertainty, representative (QUIRE, hinted), **and many more**

Active Learning by (Bandit) Learning

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

Making Active Learning Realistic

- attempt with **open-source tool** `libact`
- still **very challenging**

Thank you! Questions?