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
Active Learning

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

Hsuan-Tien Lin (NTU)  Making Active Learning More Realistic  2/22
Active Learning

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 Vicol
https://flic.kr/p/bpmGXW

Robert Vicol
https://flic.kr/p/pZv1Mf

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 $\iff$ Learning Philosophy

- **batch**: ‘duck feeding’
- **active**: ‘question asking’ (iteratively)
  —query $y_n$ of **chosen** $x_n$

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

labeled training examples

(🍎, +1), (🍎, +1), (🍎, +1)
(🍌, -1), (🍇, -1), (🍇, -1)

learning algorithm $\mathcal{A}$

final hypothesis $g \approx f$

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

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

Goal

design an algorithm that iteratively

1. **strategically query** some $\tilde{x}_s$ to get associated $\tilde{y}_s$
2. move $(\tilde{x}_s, \tilde{y}_s)$ 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?**
Strategy 1
ask most confused question

how to define confusion (uncertainty)?
Uncertainty Sampling

Uncertainty of Hard Binary Classifier

uncertain $\Leftrightarrow$ near binary decision boundary

Active Learning with SVM (Tong, 2000)

1. learn a SVM hyperplane with $\mathcal{D}_1$ (blue squares)
2. query $\tilde{x}_s$ (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**
Representative Sampling

Problem of Uncertainty Sampling

Initially

![Initial Data Points]

After many iterations

![Iterated Data Points]

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
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)

\[ \tilde{x}_s = \arg\min_{\tilde{x}} \left( \text{worst case loss on } (D_I \cup D_U) \mid (\tilde{x}, \tilde{y}) \right) \]

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 \( D_u \) part

QUIRE: promising for RS, but time consuming to compute
Representative Sampling (2/2):
Hinted Sampling

**Uncertainty Sampling (US)**

\[ \tilde{x}_s = \arg\max_{\tilde{x}} (1 - |p(\tilde{x})|) \]

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

**Hinted Sampling**

\[ \tilde{x}_s = \arg\max_{\tilde{x}} (1 - |p'(\tilde{x})|) \]

where \( p'(\tilde{x}) \approx \frac{1}{2} \) for dense regions of \( D_u \)

---

Hinted Sampling: **simple realization** of RS

---

figures from my student Chun-Liang Li’s ACML 2012 presentation (Li, 2012)
How to Query Strategically?

Strategy 1: uncertainty sampling
Strategy 2: QUIRE
Strategy 3: hinted sampling

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

do you use a fixed strategy in practice? 😊
Choice of Strategy

- **Strategy 1**: uncertainty sampling
- **Strategy 2**: QUIRE
- **Strategy 3**: 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
Choice of Strategy

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:
$A_1, A_2, \ldots, A_K$

try one strategy

“goodness” of strategy as reward

two issues: try and reward
Choice of Strategy

Reduction to Bandit

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

**gambling 😊**

- **$K$ strategies:** $A_1, A_2, \ldots, A_K$
- **try one strategy**
- **“goodness” of strategy as reward**

- **$K$ bandit machines:** $B_1, B_2, \ldots, B_K$
- **try one bandit machine**
- **“luckiness” of machine as reward**

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

intelligent choice of strategy

$\implies$ intelligent choice of **bandit machine**
Choice of Strategy

Active Learning by Learning (Hsu, 2015)

- **K** strategies: \( A_1, A_2, \ldots, A_K \)
- 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 \( A_k \) to try
2. query the \( \tilde{x}_s \) suggested by \( A_k \), and compute \( g^{(t)} \)
3. evaluate **goodness of** \( g^{(t)} \) as **reward** of trial to update EXP4.P

**only remaining problem: what reward?**
Design of Reward

### Ideal Reward

Ideal reward after updating classifier $g(t)$ by the query $(x_{nt}, y_{nt})$:

\[
\text{accuracy} \quad \frac{1}{M} \sum_{m=1}^{M} \left[ y_m = g(t)(x_m) \right] \quad \text{on test set } \{(x_m, y_m)\}_{m=1}^{M}
\]

- **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
Design of Reward

Weighted Training Accuracy as Reward

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

\[
\text{weighted training accuracy} \quad \frac{1}{t} \sum_{\tau=1}^{t} \frac{1}{p_\tau} \mathbb{I}[y_{n\tau} = g(x_{n\tau})]
\]

\[
\approx \frac{1}{N} \sum_{n=1}^{N} \mathbb{I}[y_n = g(x_n)] \quad \text{in expectation}
\]

- 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
• **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**
Realistic Active Learning

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”

Hsuan-Tien Lin (NTU)
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?