Machine Learning Approaches for Interactive Verification

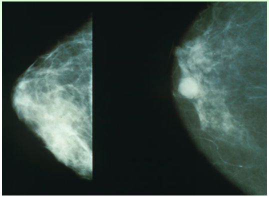
Yu-Cheng Chou and **Hsuan-Tien Lin**

Department of Computer Science & Information Engineering

National Taiwan University

May 14, 2014 (PAKDD)

Breast Cancer Screening



http://en.wikipedia.org/wiki/File:Mammo_breast_cancer.jpg

- input: X-ray images
- output: healthy (left) or breast cancer (right)
- **unbalanced**: many healthy (negative), few cancerous (positive)
- learning a good model: important
  —part of KDDCup 2008 task

to eliminate false positive:

ask human experts to verify (confirm) all positive predictions

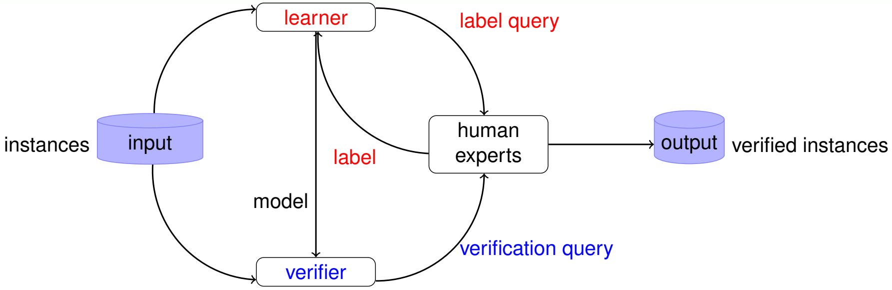
What If Too Many Positive Predictions?

if human experts cannot handle all positive instances from model

- hire more human experts (but money?)
- random sampling (but false positive?)

another possibility: 'learn' a verification assistant (verifier)

Learner versus Verifier



two stages **similarly** require human (labeling):

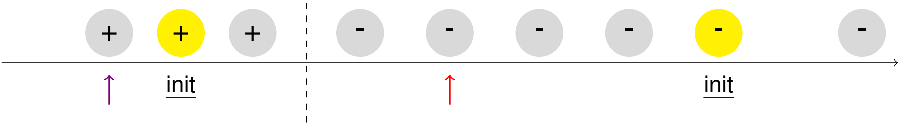
learning and verification

—save human efforts by combining the two?

Motivation: One-Dimensional Separable Data

m instances on a line

- approach 1: binary search for learning, then do verification
- approach 2: greedily do verification according to current model



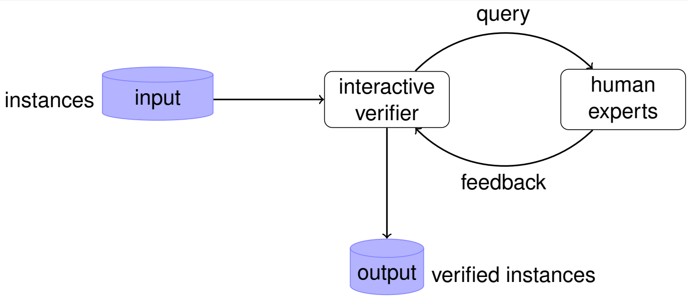
number of queries '**wasted**' on negative instances

- approach 1: $O(\log m)$
- approach 2: $O(1)$

—combining may help

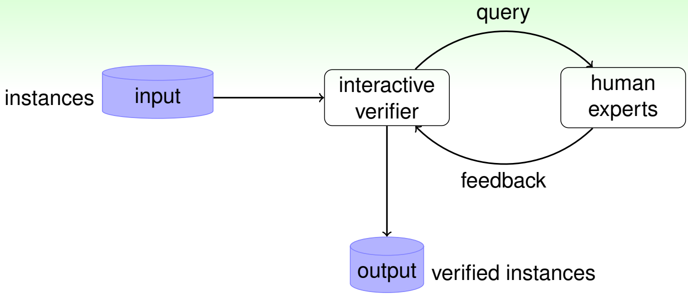
Interactive Verification Problem

- instances: $X = \{x_1, ..., x_m\}$
- **unknown** labels: $Y(x_i) \in \{-1, +1\}$
- in iteration $t = 1, 2, \cdots, T$:
  select a **different instance** q_t from X to query $Y(q_t)$



goal: maximize $\sum_{t=1}^{T} [Y(q_t) = 1]$
—verify as many positive instances as possible within T queries

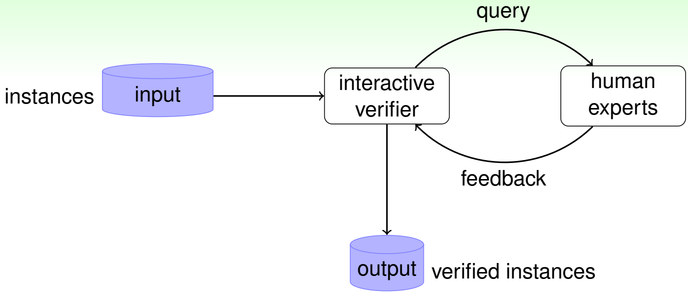
Our Contribution



an initiative to study interactive verification, which ...

- introduces a simple **framework** for designing interactive verification approaches
- **connects** interactive verification with **other related ML problems**
- exploits the connection to design **promising approaches** with **superior experimental results**

Simple Framework for Interactive Verification

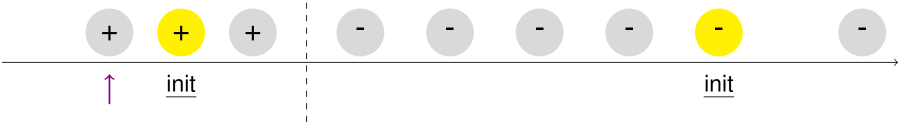


For $t = 1, ..., T$:

1. train a model by a base learner with all labeled data $(q_i, Y(q_i))$ —will consider linear SVM and denote weights by w_t
2. compute a scoring function $S(x_i, w_t)$ for each instance $x_i \in X$
3. query a different instance with highest score

different scoring functions $\Leftrightarrow$ different approaches

Greedy



- greedy: the ‘most’ positive one is the most suspicious one

$$S(x_i, w_t) = x_i^\top w_t$$

—verify greedily!

- same as approach 2 in motivating one-dimensional data

how to correct sampling bias with greedy queries?

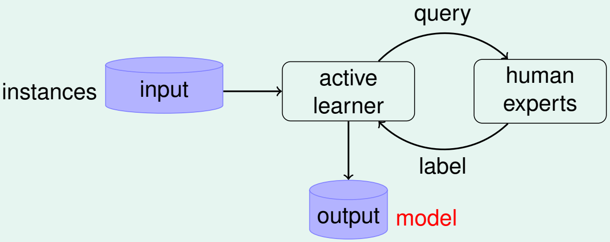
Random Then Greedy (RTG)

- random sampling for learning first
- greedy for verification later
- RTG: one-time switching with parameter ϵ

$$S(x_i, w_t) = \begin{cases} random(), & \text{if } t \leq \epsilon T \\ x_i^\intercal w_t, & \text{otherwise} \end{cases}$$

how to learn faster than random sampling?

- active learning: similar to interactive verification with different goals

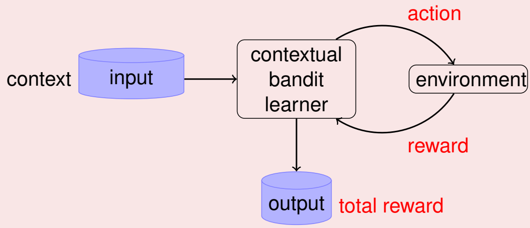


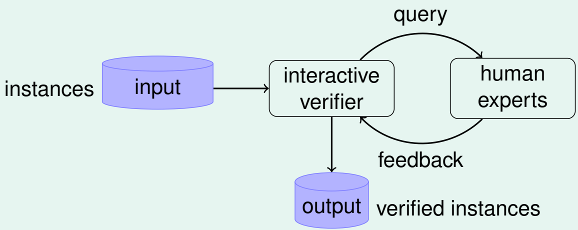
- USTG: active learning (by uncertainty sampling) first, greedy for verification later

$$S(x_i, w_t) = \begin{cases} \frac{1}{|x_i^\mathsf{T} w_t| + 1}, & \text{if } t \leq \epsilon T \\ x_i^\mathsf{T} w_t, & \text{otherwise} \end{cases}$$

how to do better than **one-time switching**?

Another Related Problem: Contextual Bandit





interactive verification
= special contextual bandit + **verified instances as rewards**

Upper Confidence Bound (UCB)

interactive verification
= special contextual bandit + **verified instances as rewards**

- contextual bandit: balance exploration (getting information) and exploitation (getting reward)
- interactive verification: balance learning and verification
- UCB: borrow idea from a popular contextual bandit algorithm

$$S(x_i, w_t) = x_i^\top w_t + \alpha \cdot \text{confidence on } x_i$$

- α: trade-off parameter between exploration (learning) and exploitation (verification)

four approaches to be studied: greedy, RTG, USTG, UCB

Comparison between the Four

	learning intent	verification intent	switching
greedy	none	positiveness	none
RTG	random sampling	positiveness	one-time
USTG	active learning	positiveness	one-time
UCB	confidence term	positiveness	dynamic

greedy: special case of RTG ($\epsilon = 0$), USTG ($\epsilon = 0$), UCB ($\alpha = 0$)

Data Sets

data set	number of instances	number of positive instances	percentage of positive instances
KDDCup2008	102294	623	0.6%
spambase	4601	1813	39.4%
a1a	1605	395	24.6%
cod-rna	59535	19845	33.3%
mushrooms	8124	3916	48.2%
w2a	3470	107	3%

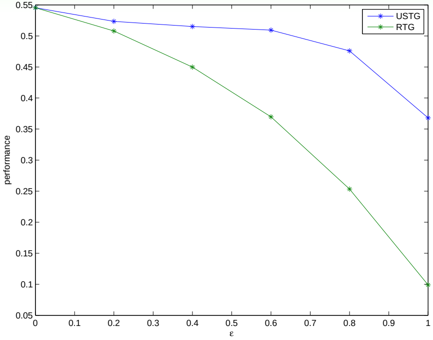
—resampled with 1000 negative and P positive instances

will show

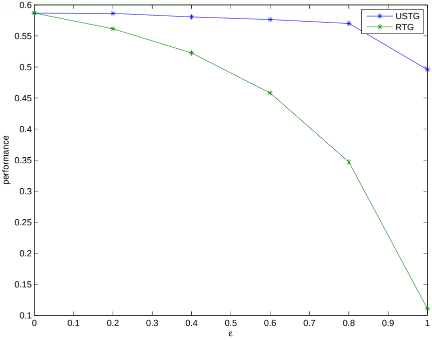
$$\frac{1}{P}\sum_{t=1}^{T}[Y(q_t) = 1]$$

under $T = 100$

Effect of ϵ



(a) KDDCup2008 with $P = 100$



(b) KDDCup2008 with $P = 50$

'naive' greedy ($\epsilon = 0$) better than RTG and USTG, why?

Good Properties of Greedy

Case 1: positive instance selected

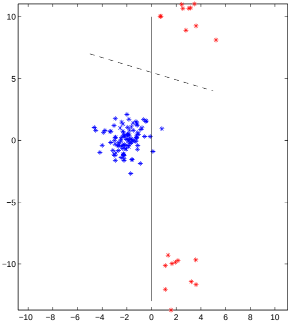
- successful verification :-)

Case 2: negative instance selected

- 'most unexpected' negative instance
- usually help learning a lot :-)

greedy approach happy ':-)' either way

Artificial Data that Fails Greedy



- two positive clusters and one big negative cluster
- greedy ignores bottom cluster: **negative instances selected doesn't help learning**

need to query 'far-away' (less confident) instances
—**UCB to the rescue**

Comparison between UCB and Greedy

$P = 50$	KDDCup2008	spambase	a1a
UCB ($\alpha = 0.2$)	0.5968 ± 0.0031	0.7306 ± 0.0020	0.3915 ± 0.0034
greedy	0.5868 ± 0.0040	0.7467 ± 0.0024	0.3883 ± 0.0034
comparison	○	×	△
$P = 50$	cod-rna	mushrooms	w2a
UCB ($\alpha = 0.2$)	0.7333 ± 0.0024	0.9776 ± 0.0007	0.6160 ± 0.0024
greedy	0.7249 ± 0.0027	0.9710 ± 0.0014	0.5944 ± 0.0030
comparison	○	○	○

UCB wins (○) often, across data sets and P

Conclusion

- **formulated a novel problem** of interactive verification
- **connected** the problem to active learning and **contextual bandit**
- **studied** a simple solution **greedy**
- **proposed** a promising solution **UCB** via contextual bandit
- **validated** that greedy and UCB lead to **promising performance**

Self-Advertisement: TAAI 2014 in Taipei

2014 Conference on Technologies and Applications of Artificial Intelligence

Submission welcomed! Thank you