Active Imitation Learning with Noisy Guidance

Kianté Brantley,[1] Amr Sharaf,[1] Hal Daumé III [1,2]
[1] University of Maryland, [2] Microsoft Research

Structured Prediction Problems

for example, Named Entity Recognition:

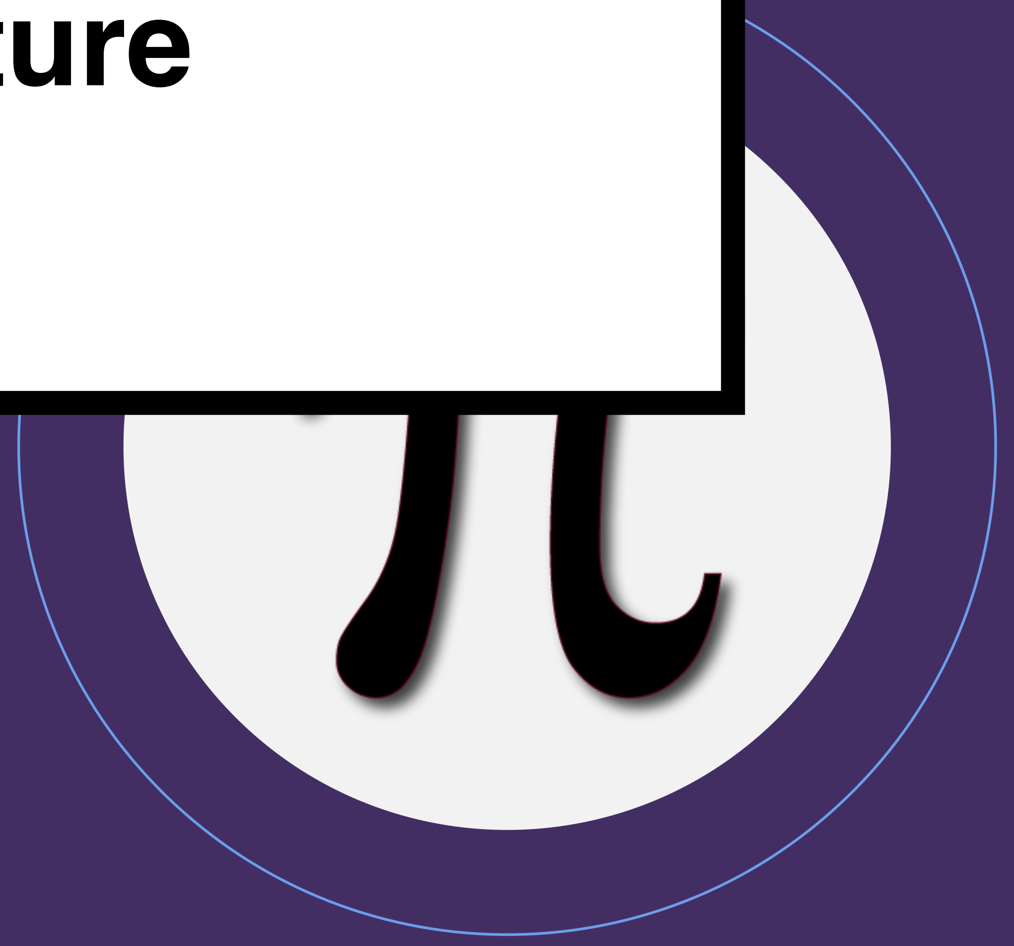
Word	Label
After	O
completing	O
his	O
Ph.D.	O
,	O
……	…..

Expert

Structured Predi

for example, Nam

Word	
After	
completing	O
his	O
Ph.D.	O
,	O
……	…..



Problem:

- Can we design an algorithm to reduce expert annotation cost for structure prediction problems?

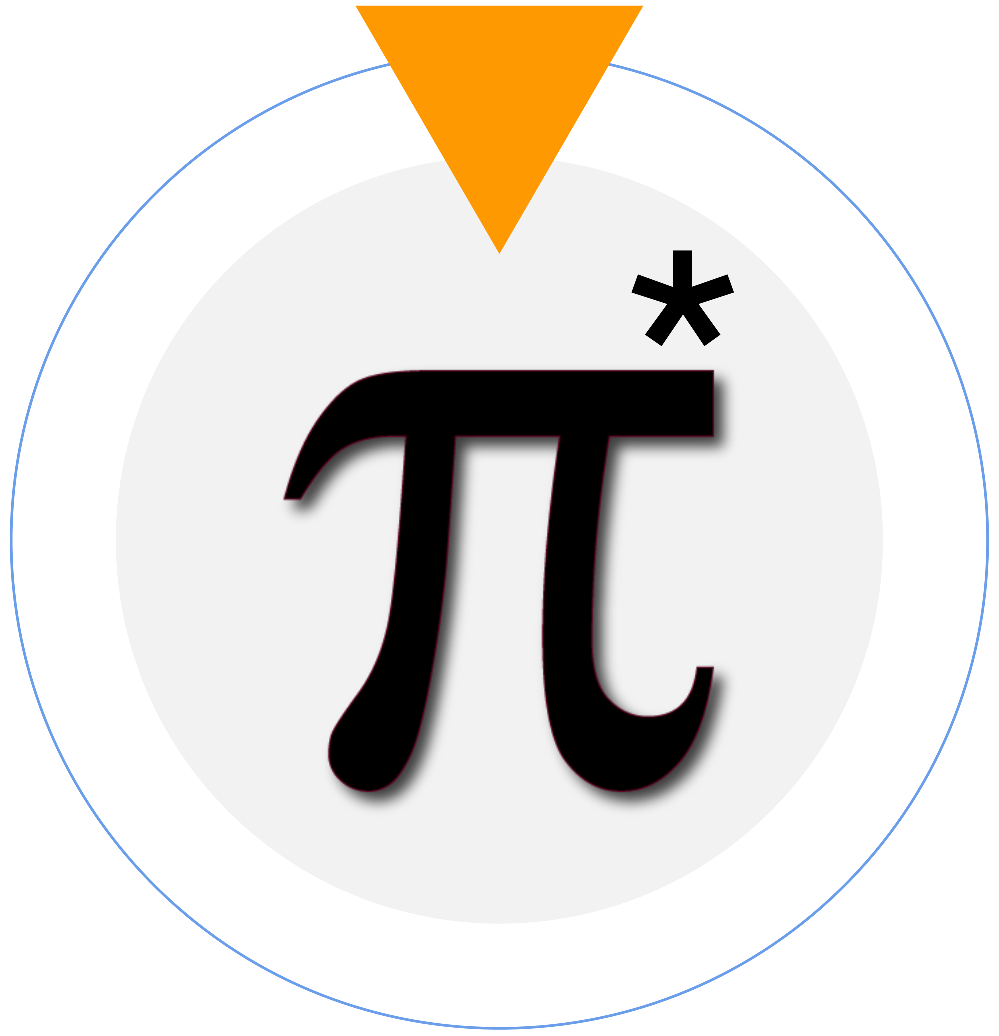
Imitation Learning

Expert Demonstrator: (Annotator)

Named Entity Recognition

Input: After completing his Ph.D. , Ellis worked at Bell Labs from 1969 to 1972 on probability theory..

Prediction: o



- **states** combine input with previous prediction
- **actions** o, per, org, misc, loc

training set: D = {(state, actions)} from expert π^*

goal: learn agent π_θ(s) -> a

Imitation Learning using DAgger

Initialize Dataset D

Initialize $\hat{\pi}_1$

for $i = 1$ to N do

$\pi_i = \beta_i \pi^* + (1 -$

Sample T-step tra

Get dataset $D_i =$

Aggregate datase

Train classifier $\hat{\pi}$

Pro:

- The policy is able to learn from its own state distribution.

Named Entity Recognition

npleting his Ph.D., Ellis worke

O O O O PER O

O O O O PER O

Stéphane Ross, Geoff J. Gordon, and J. Andrew Bag- nell. 2011. A reduction of imitation learning and structured prediction to no-regret online learning. In AI-Stats.

Imitation Learning using DAgger

Initialize Dataset D

Initialize $\hat{\pi}_1$

for $i = 1$ to

$\pi_i = \beta_i \pi^* +$

Sample T-s

Get datase

Aggregate

Train classifier $\hat{\pi}_{i+1}$ on D

Con:

- For every state that we visited we queried an expert for the optimal action.

Name Entity Recognition

fter completing his Ph.D., Ellis worke

O O O O O PER O

O O O O O PER O

Stéphane Ross, Geoff J. Gordon, and J. Andrew Bag- nell. 2011. A reduction of imitation learning and structured prediction to no-regret online learning. In AI-Stats.

Active Learning

Key Idea: The learner queries the expert for labels — only when it is uncertain

Formally

for each trial $t = 1,2,...$

  observe instance $x_t \in \mathbb{R}$

  set $\hat{p}_t = \pi_\theta(y_t^1 \mid x_t) - \pi_\theta(y_t^2 \mid x_t)$ (Margin between the most likely and the second most likely labels)

  predict with $\hat{y}_t = \textbf{argmax}(\pi_\theta)$

  draw a Bernoulli variable Z_t **of parameter** $\dfrac{b}{b + |\hat{p}_t|}$ (Confidence parameter b)

  if $Z_t = 1$

    query label y_t **and perform update**

[T. Scheffer, C. Decomain, and S. Wrobel. Active hidden Markov models for information extraction. Proceedings of the International Conference on Advances in Intelligent Data Analysis (CAIDA), 2001.]

[Nicolò Cesa-Bianchi, Claudio Gentile, and Luca Zaniboni. 2006. Worst-case analysis ofselective sampling for linear classification. JMLR.]

Leveraging **Active Learning**

Key Idea: The learner queries the expert for labels — only when it is uncertain

Formally

for each trial $t = 1,2,...$

observe instance $x_t \in \mathbb{R}$

set $\hat{p}_t = \pi_\theta(y_t^1 \,|\, x_t) - \pi_\theta(y_t^2 \,|\, x_t)$ (Margin)

predict with $\hat{y}_t = \textbf{argmax}(\pi_\theta)$

draw a Bernoulli variable Z_t **of parameter** $\frac{\quad}{b + |\hat{p}_t|}$ (Confidence parameter b)

if $Z_t = 1$

query label y_t **and perform update**

Confidence parameter: b

- ❑ **big - increases the probability of requesting a label**
- ❑ **small - decreases the probability of requesting a label**

[T. Scheffer, C. Decomain, and S. Wrobel. Active hidden Markov models for information extraction. Proceedings of the International Conference on Advances in Intelligent Data Analysis (CAIDA), 2001.]

[Nicolò Cesa-Bianchi, Claudio Gentile, and Luca Zaniboni. 2006. Worst-case analysis ofselective sampling for linear classification. JMLR.]

Active Learning with DAgger

Initialize Dataset D

Initialize $\hat{\pi}_1$

for $i = 1$ to N do

$\pi_i = \beta_i \pi^* + (1 - \beta_i)\hat{\pi}$

Sample T-step traject

for $t = 1$ to T

set $\hat{p}_t = \pi_\theta(y_t^1 \mid s_t$

draw Bernoulli variable Z_t of parameter $b + |\hat{p}_t|$

if $Z_t = 1$

Get dataset $D_t = \{(s_t, \pi^*(s_t))\}$

Aggregate dataset $D \leftarrow D \cup D_t$

Train classifier $\hat{\pi}_{i+1}$ on D

Question:

- [ ] Can reduce expert queries even further?

gnition

leting his Ph.D., Ellis worke

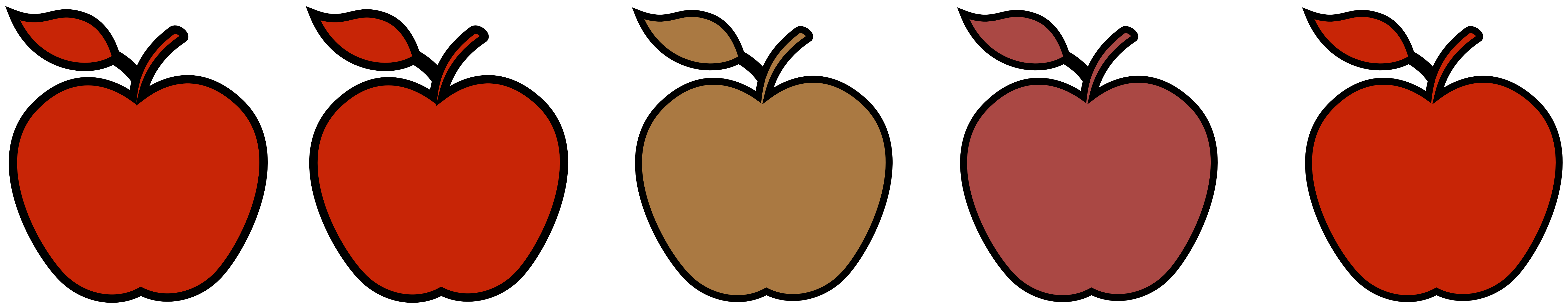
O O O O PER O

PER O

Our Approach: LeaQI
(Learning to Query for Imitation)

Key Ideas:

- We assume access to a noisy heuristic function
- Use a disagreement classifier to decide if we should query the expert or the heuristic function
- Train the disagreement classifier using the Apple Tasting framework



Apple Tasting Framework

One-Side Feedback Problem

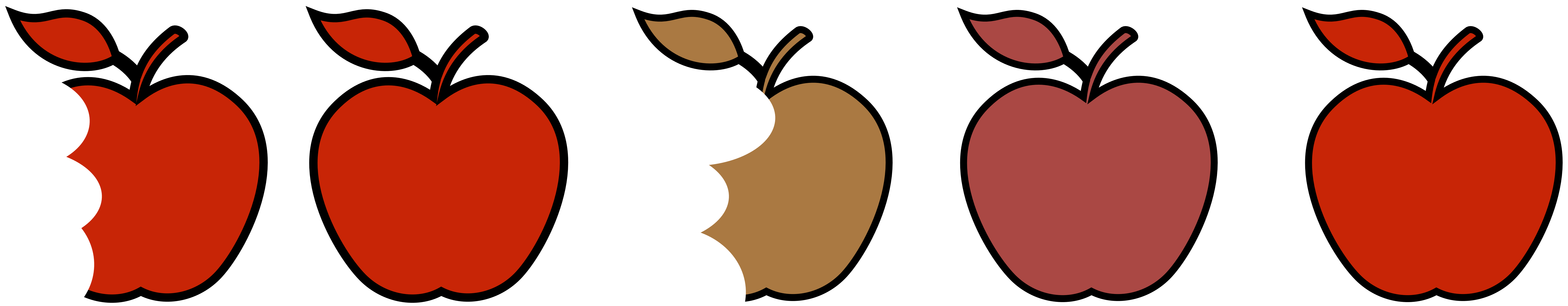
Learner encounters apples one by one

Goal is to avoid tasting to many bad apples and avoid throwing away to many good apples (reduce false negative rates)

Problem is the learner can only identify the good and bad apples by tasting them

Learner only gets feedback for apples that it tastes

Learner does not feedback for apples that it throws away



One-Sided Feedback Learning

Named Entity Recognition

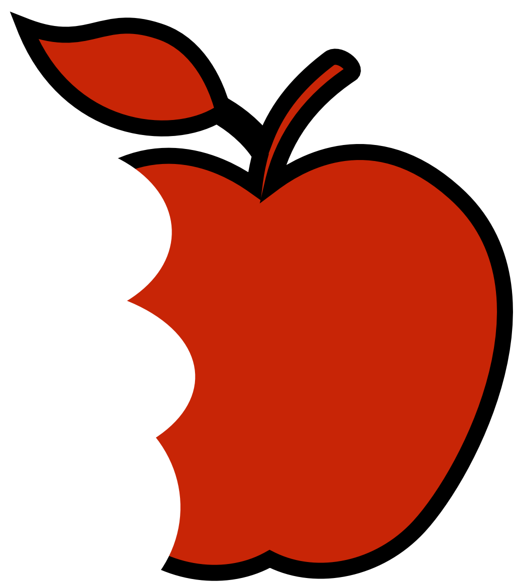
Input: After completing his Ph.D. , Ellis worked at Bell Labs from 1969 to 1972 on pro

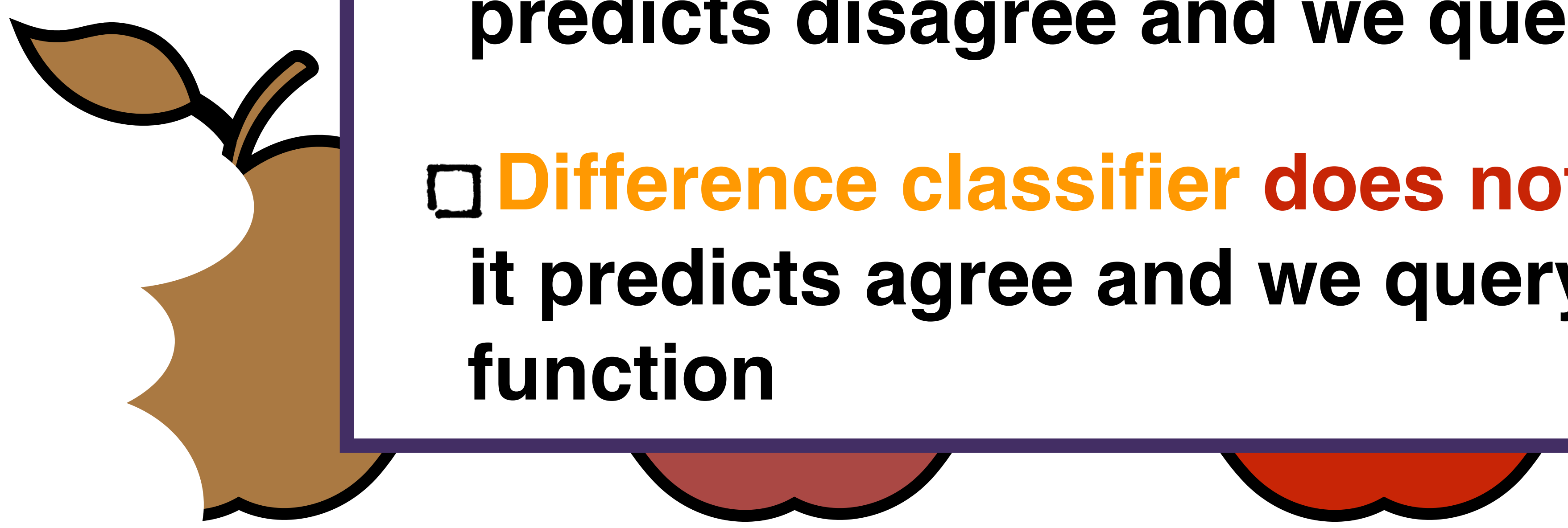
π

Heuristic Function

Gazetteer: O O O

π^* O O O





Heuristic Function

- Noisy, bias and cheap

LeaQI One-Side Feedback Problem

- Learn difference classifier to predict when a Heuristic and Expert disagree
- Difference classifier only gets feedback when it predicts disagree and we query the expert
- Difference classifier does not get feedback when it predicts agree and we query the heuristic function

LeaQI

draw Bernoulli variable Z_t of parameter $\frac{b}{b + |\hat{p}_t|}$

if $Z_t = 1$

$\hat{d}_i = h_i(s)$ Set difference classifier

If AppleTaste(s , $\pi^h(s)$, $\hat{d}_i$)

Aggregate dataset $D \leftarrow D \cup \{(s, \pi^h(s))\}$

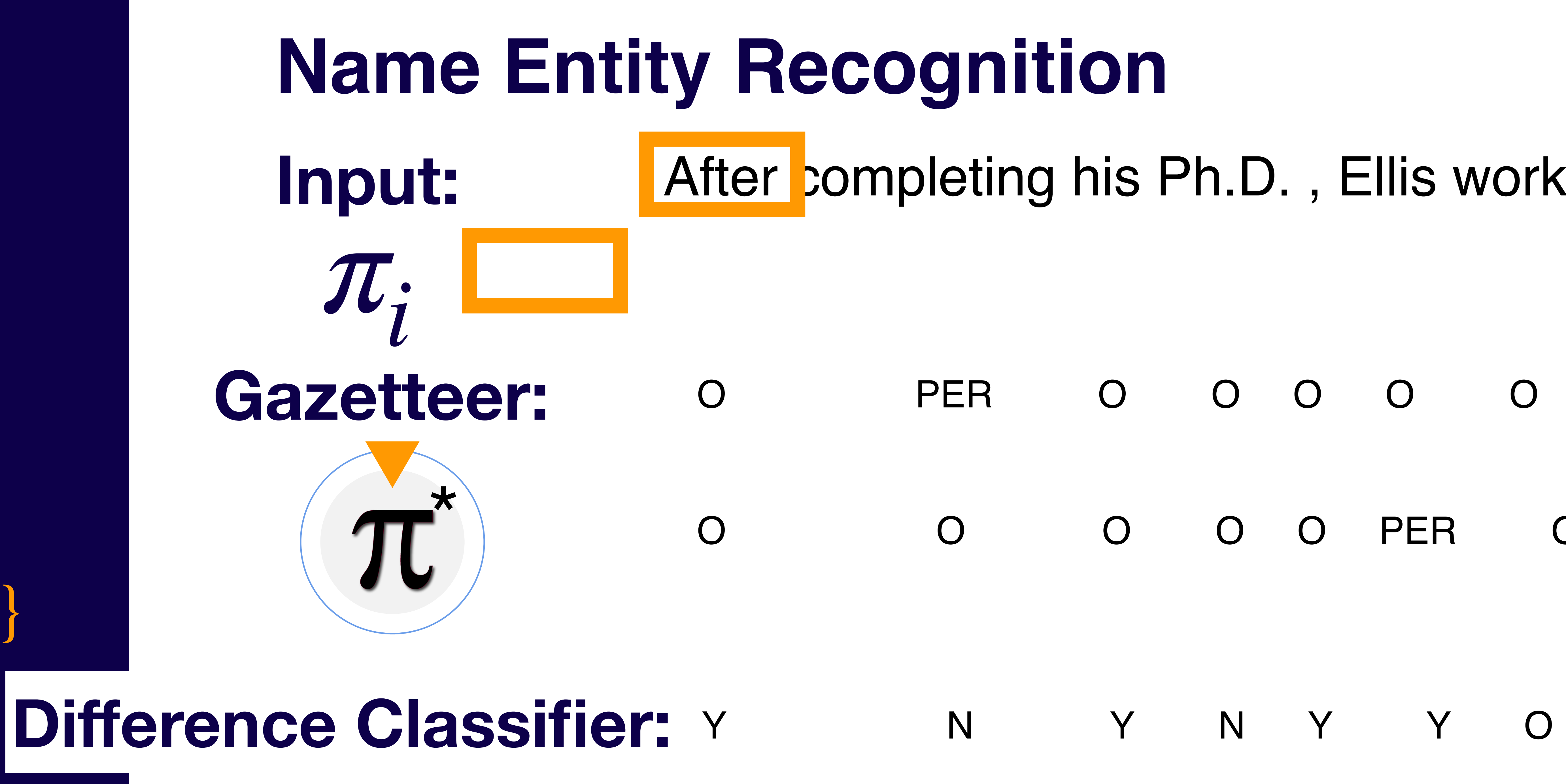
else

Aggregate dataset $D \leftarrow D \cup \{(s, \pi^*(s))\}$

Aggregate dataset $S \leftarrow S \cup \{(s, \pi^h(s), \hat{d}, d)\}$

Train classifier $\hat{\pi}_{i+1}$ on D

Train difference classifier h_{i+1} on S



Experiment Details

	NER	Keyphrase	POS
Language	English	English	Modern Greek
Dataset	CoNLL'03	SemEval 2017 Task 10	Universal Dependencies
Heuristic	Gazeteer	Unsupervised model	Dictionary Wiktionary
Huer. Quality	P88%, R27%	P20%, R44%	67% acc

Q1

Active vs Passive

Q2

Heuristic as features vs Policy

Q3

Difference Classifier Efficacy

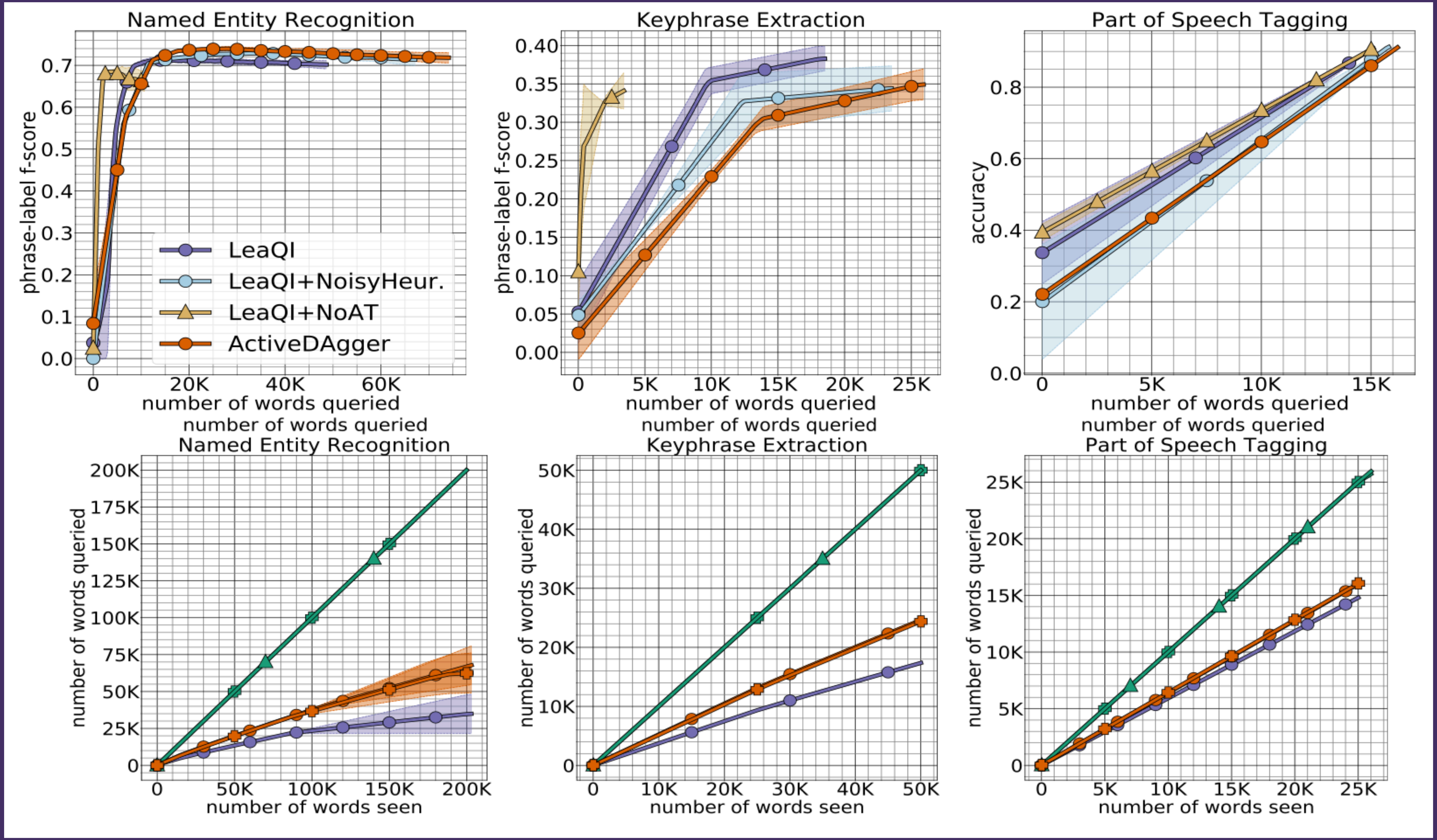
Q4

Apple Tasting Efficacy

Q5

Robustness to Poor a Heuristic

Experiment Results



- We showed that the Apple Tasting framework has practical benefits
- We showed a relationship between using a heuristic function and One-side feedback learning
- We introduced a new algorithm and evaluated it on 3 task

Thank you!