UNIVERSITÄT HEIDELBERG
ZUKUNFT SEIT 1386

Improving a Neural Semantic Parser by Counterfactual Learning from Human Bandit Feedback

Carolin Lawrence, Stefan Riezler

Heidelberg University
Institute for Computational Linguistics

July 17, 2018

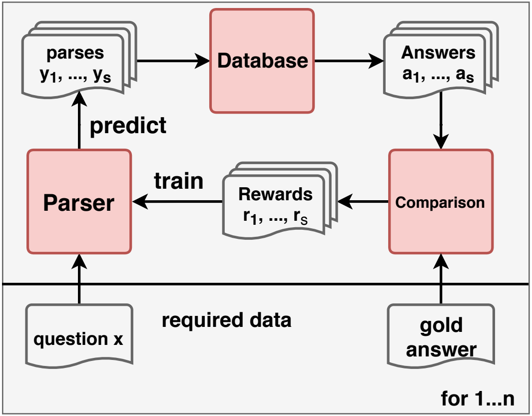
UNIVERSITÄT HEIDELBERG ZUKUNFT SEIT 1386

Situation Overview

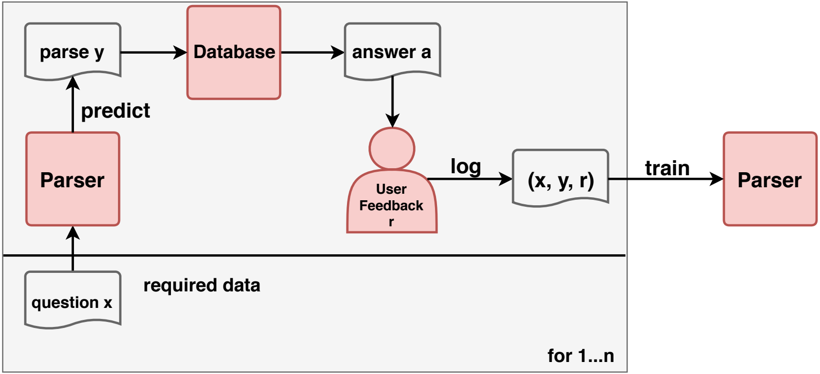
- Situation: deployed system (e.g. QA, MT ...)
- Goal: improve system using human feedback
- Plan: create a log $\mathcal{D}_{log}$ of user-system interactions
  & improve system offline (safety)

Here: Improve a Neural Semantic Parser

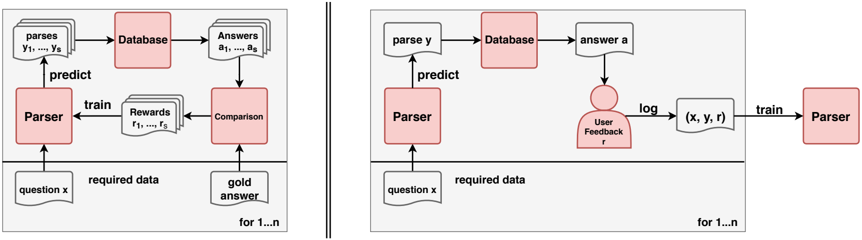
Contrast to Previous Approaches



Our Approach



Our Approach



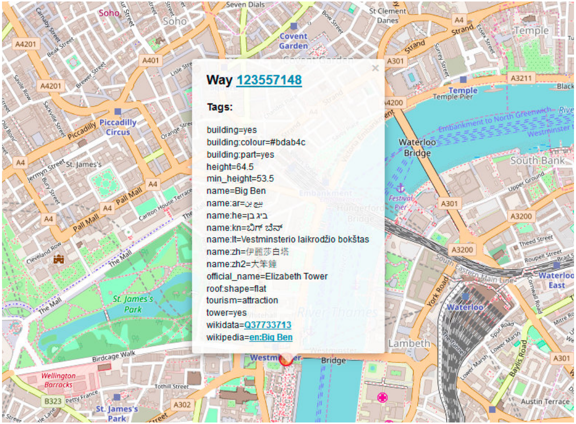
- No supervision: given an input, the gold output is unknown
- Bandit: feedback is given for only one system output
- Bias: log $\mathcal{D}$ is biased to the decisions of the deployed system

Solution: Counterfactual / Off-policy Reinforcement Learning

Task

A natural language interface to OpenStreetMap

- OpenStreetMap (OSM): geographical database
- NLMAPS V2: extension of the previous corpus, now totalling 28,609 question-parse pairs



A natural language interface to OpenStreetMap

- example question: "*How many hotels are there in Paris?*"
  Answer: 951
- correctness of answers are difficult to judge
  $\rightarrow$ judge parses by making them human-understandable
- feedback collection setup:
  1. automatically convert a parse to a set of statements
  2. humans judge the statements

Example: Feedback Formula

Question #216: **What are the names of cinemas that are within walking distance from the Place de la République in Paris?**

		Information found in Question?
Town	Paris	**Yes** / No
Reference Point	name : Place de la République	**Yes** / No
POI(s)	amenity : parking	Yes / **No**
Question Type	What's the name	**Yes** / No
Proximity	Around/Near	**Yes** / No
Distance	Walking distance	**Yes** / No

Submit

query(around(center(area(keyval('name','Paris')), nwr(keyval('name','Place de la République'))), search(nwr(keyval('amenity','parking'))), maxdist(WALKING_DIST)),qtype(findkey('name')))

Objectives

UNIVERSITÄT HEIDELBERG ZUKUNFT SEIT 1386

Counterfactual Learning

Resources

collected log $\mathcal{D}_{log} = \{(x_t, y_t, \delta_t)\}_{t=1}^{n}$ with

- x_t: input
- y_t: most likely output of deployed system π_0
- $\delta_t \in [-1, 0]$: loss (i.e. negative reward) received from user

Deterministic Propensity Matching (DPM)

- minimize the expected risk for a target policy π_w

$$\hat{R}_{\text{DPM}}(\pi_w) = \frac{1}{n} \sum_{t=1}^{n} \delta_t \pi_w(y_t | x_t)$$

- improve π_w using (stochastic) gradient descent
- high variance $\rightarrow$ use multiplicative control variate

Multiplicative Control Variate

UNIVERSITÄT HEIDELBERG ZUKUNFT SEIT 1386

- for random variables X and Y, with $\bar{Y}$ the expectation of Y:

$$\mathbb{E}[X] = \mathbb{E}[\frac{X}{Y}] \cdot \bar{Y}$$

$\rightarrow$ RHS has lower variance if Y positively correlates with X

DPM with Reweighting (DPM+R)

$$\hat{R}_{\text{DPM+R}}(\pi_w) = \frac{\frac{1}{n}\sum_{t=1}^{n} \delta_t \pi_w(y_t|x_t)}{\frac{1}{n}\sum_{t=1}^{n} \pi_w(y_t|x_t)} \cdot 1 \qquad \text{Reweight Sum } R$$

- reduces variance but introduces a bias of order $O(\frac{1}{n})$ that decreases as n increases
  $\rightarrow$ n should be as large as possible
- Problem: in stochastic minibatch learning, n is too small

One-Step Late (OSL) Reweighting

UNIVERSITÄT HEIDELBERG ZUKUNFT SEIT 1386

Perform gradient descent updates & reweighting asynchronously

- evaluate reweight sum R on the entire log of size n using parameters w'
- update using minibatches of size m, $m \ll n$
- periodically update R

$\rightarrow$ retains all desirable properties

DPM+OSL

$$\hat{R}_{\text{DPM+OSL}}(\pi_w) = \frac{\frac{1}{m}\sum_{t=1}^{m} \delta_t \pi_w(y_t|x_t)}{\frac{1}{n}\sum_{t=1}^{n} \pi_{w'}(y_t|x_t)}$$

Token-Level Feedback

DPM+T

$$\hat{R}_{\text{DPM+T}}(\pi_w) = \frac{1}{n} \sum_{t=1}^{n} \left(\prod_{j=1}^{|y|} \delta_j \pi_w(y_j | x_t) \right)$$

DPM+T+OSL

$$\hat{R}_{\text{DPM+T+OSL}}(\pi_w) = \frac{\frac{1}{m} \sum_{t=1}^{m} \left(\prod_{j=1}^{|y|} \delta_j \pi_w(y_j | x_t) \right)}{\frac{1}{n} \sum_{t=1}^{n} \pi_{w'}(y_t | x_t)}$$

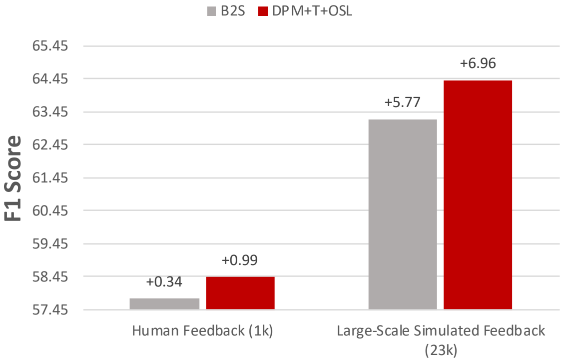
Experiments

Experimental Setup

- sequence-to-sequence neural network NEMATUS
- deployed system: pre-trained on 2k question-parse pairs
- feedback collection:
  1. humans judged 1k system outputs
     - average time to judge a parse: 16.4s
     - most parses (>70%) judged in <10s
  2. simulated feedback for 23k system outputs
     - token-wise comparison to gold parse
- bandit-to-supervised conversion (B2S): all instances in log with reward 1 are used as supervised training

UNIVERSITÄT HEIDELBERG ZUKUNFT SEIT 1386

Experimental Results



Take Away

UNIVERSITÄT HEIDELBERG ZUKUNFT SEIT 1386

Counterfactual Learning

- safely improve a system by collecting interaction logs
- applicable to any task if the underlying model is differentiable
- DPM+OSL: new objective for stochastic minibatch learning

Improving a Semantic Parser

- collect feedback by making parses human-understandable
- judging a parse is often easier & faster than formulating a parse or answer

NLmaps v2

- large question-parse corpus for QA in the geographical domain

Future Work

- integrate feedback form in the online NL interface to OSM