Introduction into Imitation Learning

Artem Sokolov

Institute for Computational Linguistics, Heidelberg University

8 October 2018

1 Admin
2 Literature
3 What is Imitation Learning?
4 Applications of Imitation Learning
5 Problems with Reinforcement Learning
6 Problems with Supervised Learning
7 What does Imitation Learning add?
Admin
Before we start

- lots of reading
- knowledge of these is helpful:
  - foundations of probability theory
  - foundations of statistical machine learning
  - reinforcement learning
  - structured prediction
  - neural networks
- assessment: presentations or projects
- projects: programming (if you want to really learn)
  - project emphasis open applied research problems
This a block module, so **you have two options:**

1. **presentation**
   - presentation (end of the 1st or in the 2nd week)
   - write-up detailing the presented article, during the semester
   - compare with rival approaches
   - describe and defend extensions if any
   - small implementation is encouraged but not required
This a block module, so **you have two options:**

1. **presentation**
   - presentation (end of the 1st or in the 2nd week)
   - write-up detailing the presented article, during the semester
   - compare with rival approaches
   - describe and defend extensions if any
   - small implementation is encouraged but not required

2. **project**
   - briefly describe the task, research question and your plan (2nd week)
   - no slides necessary (unless you prefer)
   - implementation at your own pace during the semester
   - report detailing the approach and experiments
   - bi-weekly updates over skype or email
■ presentation:
  ▸ larger up-front effort
  ▸ mostly free for the rest of semester
- **presentation:**
  - larger up-front effort
  - mostly free for the rest of semester

- **project:**
  - lesser effort now
  - more time to think and more experience gained
Deliverables

- presentation:
  - larger up-front effort
  - mostly free for the rest of semester

- project:
  - lesser effort now
  - more time to think and more experience gained

- both projects and presentations are open-ended
  - no fixed upper bar of what can be done
  - possible that some things may not work
  - you may be asked to add/change things as you advance
Deliverables

- presentation:
  - larger up-front effort
  - mostly free for the rest of semester

- project:
  - lesser effort now
  - more time to think and more experience gained

- both projects and presentations are open-ended
  - no fixed upper bar of what can be done
  - possible that some things may not work
  - you may be asked to add/change things as you advance

- there is a list of tentative projects and presentation topics

- but you are encouraged to come up with your own ideas
1st week, Oct. 8-12
- 10:15-11:45 morning session
- 13:15-14:45 afternoon session
- first article presentations
- no morning session on Friday - Erstifrühstück

Till Sat Oct. 13 23:59
- inform via email whether you do a presentation or a project
2nd week, Oct. 15-19
- article presentations and project Q&A
- consultations on the articles/projects

rest of the semester
- work on projects and writeups
- updates via email/skype
- a detailed explanation of a state-of-the-art method
- or it’s algorithmic (or theoretical or heuristic) extension
- a detailed explanation of a state-of-the-art method
- or it’s algorithmic (or theoretical or heuristic) extension

**Setup:**
- 1 person
- a detailed explanation of a state-of-the-art method
- or it’s algorithmic (or theoretical or heuristic) extension

**Setup:**

- 1 person
- **task:**
  - pick an article from the list or propose your own
  - present the approach, include the glossed-over parts
  - try to formulate possible extensions
  - answer additional questions
- a detailed explanation of a state-of-the-art method
- or it’s algorithmic (or theoretical or heuristic) extension

**Setup:**

- 1 person
- **task:**
  - pick an article from the list or propose your own
  - present the approach, include the glossed-over parts
  - try to formulate possible extensions
  - answer additional questions

- **timeline**
  - before Oct. 13: decide on the article
  - Oct. 15-18: time to discuss/ask questions
  - Oct. 15-19: in-class presentation
  - (except 3 easier articles that, if selected, should be presented this week)
  - end of semester: final write-up due
    - try to formulate and defend possible extensions
    - include analysis of extensions if any
    - comparison to other methods and limitations
an implementation of a state-of-the-art algorithm for an NLP task
preferably MT
- an implementation of a state-of-the-art algorithm for an NLP task
- preferably MT

**Setup:**
- 1-3 persons (expectations scale accordingly)
an implementation of a state-of-the-art algorithm for an NLP task
preferably MT

Setup:
- 1-3 persons (expectations scale accordingly)
- task:
  - pick a project from a list or propose your own
  - implement it
  - formulate and try out possible extensions
  - compare to reasonable baselines
an implementation of a state-of-the-art algorithm for an NLP task
preferably MT

Setup:
1-3 persons (expectations scale accordingly)
task:
- pick a project from a list or propose your own
- implement it
- formulate and try out possible extensions
- compare to reasonable baselines
timeline
- before Oct. 13: decide on the project
- Oct. 15-19: time to discuss/ask questions
- Oct. 15-20: in-class project discussion (no slides necessary)
- semester: bi-weekly updates if needed
- end of semester: final report due
  - include all the experimental results and link to the code
  - structure like a conference paper (setting, prior work, why it matters, your approach, results, their analysis, limitations, future directions).
presentation
  ➞ 70% - quality of the presentation and extensions
  ➞ 30% - quality of the final write-up

project
  ➞ 70% - quality of the implementation and results
  ➞ 30% - quality of the final report
• presentation
  ➔ 70% - quality of the presentation and extensions
  ➔ 30% - quality of the final write-up

• project
  ➔ 70% - quality of the implementation and results
  ➔ 30% - quality of the final report

• more points for novelty and your ideas
presentation
  - 70% - quality of the presentation and extensions
  - 30% - quality of the final write-up

project
  - 70% - quality of the implementation and results
  - 30% - quality of the final report

more points for novelty and your ideas

the effort is what mainly counts
After this course you should be able to:

- recognize tasks solvable with imitation learning
- map NLP and structured prediction problems to imitation learning
- understand deficiencies of some straight-forward approaches to IL
- solve problems with IL
reinforcement learning

- Sutton and Barto, "Reinforcement Learning", 2nd edition 2018
  https://sites.ualberta.ca/~szepesva/RLBook.html
reinforcement learning
- Sutton and Barto, "Reinforcement Learning", 2nd edition 2018
  https://sites.ualberta.ca/~szepesva/RLBook.html

imitation learning
- Attia and Dayan, "Global overview of Imitation Learning", 2018
  https://arxiv.org/abs/1801.06503
- Daumé III, "A Course in Machine Learning", Chapter 18
- Osa et al. "An Algorithmic Perspective on Imitation Learning", only intro, 2017
some good tutorials:

- Hal Daumé III, "From Structured Prediction to Inverse Reinforcement Learning", ACL’05
  users.umiacs.umd.edu/~hal/SPIRL/10-07-acl-spirl.pdf
- Hal Daumé III and John Langford, "Learning to Search for Joint Prediction", ICML/NAACL’16
  http://hunch.net/~l2s
- Yisong Yue and Hoang M. Le "Imitation Learning", ICML’18
  sites.google.com/view/icml2018-imitation-learning
- Johannes Heidecke, "Inverse Reinforcement Learning", 2018
  thinkingwires.com/posts/2018-02-13-irl-tutorial-1.html
- some good tutorials:
  - Hal Daumé III, "From Structured Prediction to Inverse Reinforcement Learning", ACL’05
    users.umiacs.umd.edu/~hal/SPIRL/10-07-acl-spirl.pdf
  - Hal Daumé III and John Langford, "Learning to Search for Joint Prediction", ICML/NAACL’16
    http://hunch.net/~l2s
  - Yisong Yue and Hoang M. Le "Imitation Learning", ICML’18
    sites.google.com/view/icml2018-imitation-learning
  - Johannes Heidecke, "Inverse Reinforcement Learning", 2018
    thinkingwires.com/posts/2018-02-13-irl-tutorial-1.html

- neural networks
    https://arxiv.org/abs/1703.01619
  - NLP: Yoav Goldberg, "A Primer on Neural Network Models for Natural Language Processing", 2015
    https://arxiv.org/abs/1510.00726
What is Imitation Learning?
The purpose of imitation learning is to efficiently learn a desired behavior by imitating an expert's behavior.
The purpose of imitation learning is to efficiently learn a desired behavior by imitating an expert's behavior.

Sounds familiar..
The purpose of imitation learning is to efficiently learn a desired behavior by imitating an expert's behavior.

Sounds familiar..

- is it supervised learning?
The purpose of imitation learning is to efficiently learn a desired behavior by imitating an expert's behavior.

Sounds familiar..

- Is it reinforcement learning?
The purpose of imitation learning is to efficiently learn a desired behavior by imitating an expert's behavior.

Imitation learning is a fusion between supervised learning and reinforcement learning:
In the order of goal specification:

- reinforcement learning
  (weak: no explicit goals, but intermediate rewards)

- imitation learning
  (stronger: no explicit goals, but some examples how to reach them)

- supervised learning
  (full: explicit goals even for intermediate steps)

- unsupervised learning
  (none: no goals, no rewards)
In the order of goal specification:

- reinforcement learning
  (weak: no explicit goals, but intermediate rewards)

- supervised learning
  (full: explicit goals even for intermediate steps)
In the order of goal specification:

- reinforcement learning (weak: no explicit goals, but intermediate rewards)
- supervised learning (full: explicit goals even for intermediate steps)
- unsupervised learning (none: no goals, no rewards)
In the order of goal specification:

- **reinforcement learning**  
  (weak: no explicit goals, but intermediate rewards)

- **imitation learning**  
  (stronger: no explicit goals, but some examples how to reach them)

- **supervised learning**  
  (full: explicit goals even for intermediate steps)

- **unsupervised learning**  
  (none: no goals, no rewards)
In the order of goal specification:

- **reinforcement learning**  
  (weak: no explicit goals, but intermediate rewards)

- **imitation learning**  
  (stronger: no explicit goals, but some examples how to reach them)

- **supervised learning**  
  (full: explicit goals even for intermediate steps)

- **unsupervised learning**  
  (none: no goals, no rewards)

In the view of different supervision, it makes sense to develop dedicated methods.
Applications of Imitation Learning
- reinforcement learning – impressive but relatively few successes in unrestricted environments
  - board/computer games, robotics
  - except: bandit learning really shines e.g. in ad placement, recommendation

- supervised learning – overwhelming majority of all successes of machine learning
  - vision, music, NLP, ...
  - all kinds of predictive data analysis

- unsupervised learning – there is progress, but it has yet to catch up
reinforcement learning – impressive but relatively few successes in unrestricted environments
- board/computer games, robotics
- except: bandit learning really shines e.g. in ad placement, recommendation

imitation learning – borrow concepts from reinforcement learning while not throwing away supervised learning
- also games (many RL successes are due to imitation learning components)
- navigation
- self-driving cars

supervised learning – overwhelming majority of all successes of machine learning
- vision, music, NLP, ...
- all kinds of predictive data analysis

unsupervised learning – there is progress, but it has yet to catch up
Successful applications

- self-driving vehicles
- games and bots
- website optimization
- structured prediction and **NLP**
Several ways to argue why we need IL:

- supervision strength
- avoiding some of hurdles in reinforcement learning
- relaxing assumptions of supervised learning
- sample and time efficiency
- deployment safety
Problems with Reinforcement Learning
Why not just use RL?
WHENEVER SOMEONE ASKS ME IF RL WORKS, I TELL THEM IT DOESN'T AND 70% OF THE TIME, I'M RIGHT

[Irpan'18]
[Ng&Russel'00]

The entire field of reinforcement learning is founded on the presupposition that the reward function, . . . is the most succinct, robust, and transferable definition of the task.
[Ng&Russel’00]

The entire field of reinforcement learning is founded on the presupposition that the reward function, . . . is the most succinct, robust, and transferable definition of the task.

- reward specification, which is often hard to specify exactly
  - unless we set the rules of the environment (games!)
- even good reward functions can be gamed
- RL is often sample inefficient
- reproducability
- often cannot allow unrestricted exploration
Example: Misspecification of RL rewards

- goal: finish the race
- rewards given also for collecting powerups to the race faster
- farming the powerups gives more points than finishing the race!
Amazon/HDU bandit MT learning competition:

- task: adapt to a new domain only using BLEU scores on own translations
- reward: BLEU (sentence level)
- evaluation: BLEU (corpus level)
Example: Gaming good RL rewards

Amazon/HDU bandit MT learning competition:

- task: adapt to a new domain only using BLEU scores on own translations
- reward: BLEU (sentence level)
- evaluation: BLEU (corpus level)

Different winners depending on what you look at
Amazon/HDU bandit MT learning competition:

- **task**: adapt to a new domain only using BLEU scores on own translations
- **reward**: BLEU (sentence level)
- **evaluation**: BLEU (corpus level)
Abstractive summarization with the ROUGE score as reward

[Paulus, 2017]

Button was denied his 100th race for McLaren after an ERS prevented him from making it to the start-line. It capped a miserable weekend for the Briton. Button has out-qualified. Finished ahead of Nico Rosberg at Bahrain. Lewis Hamilton has. In 11 races. . The race. To lead 2,000 laps. . In. . . And.
Amazon/HDU bandit MT learning competition:

1M sentences to learn how to translate in a new domain
humans would learn it from few dozens of examples
Example: RL is sample-inefficient

- Atari games
- 83 hours of play for RL
- humans learn it in minutes
[Weiner’60]

If we use, to achieve our purposes, a mechanical agency with whose operation we cannot efficiently interfere once we have started it, because the action is so fast and irrevocable that we have not the data to intervene before the action is complete, then we had better be quite sure that the purpose put into the machine is the purpose which we really desire...
A machine is not a genie, it does not work by magic, [...] The “intentions” which the machine seems to manifest are the intentions of the human programmer, as specified in advance, ... To believe otherwise is either to believe in magic [...]

[Samuel’60]
A machine is not a genie, it does not work by magic, [...] The "intentions" which the machine seems to manifest are the intentions of the human programmer, as specified in advance, ... To believe otherwise is either to believe in magic [...] An apparent exception to these conclusions might be claimed for projected machines of the so-called "neural net" type [...] Since the internal connections would be unknown, the precise behavior of the nets would be unpredictable and, therefore, potentially dangerous.
Deep RL is popular because it’s the only area in ML where it’s socially acceptable to train on the test set.

- exaggeration, but partially true
- raises related concerns for conventional supervised learning
Problems with Supervised Learning
By conventional ML we mean supervised batch-learning

- relies on the hard-to-guarantee i.i.d. assumptions
- when used blindly on non-i.i.d data can go wrong
- slow inference for structured SL
- uncertainty guarantees
Convenient paradigm, but

- training data often will not contain important cases
- big data: no-one checks the validity of i.i.d
- in multi-step decision processes the data distribution is dependent on the agent
Example: non-i.i.d and missing data
Convenient paradigm, but despite it’s invalidness:
- supervised learning is often works for IL!
- is very simple
- always should be tried first
What does Imitation Learning add?
Imitation learning is a fusion between supervised learning and reinforcement learning:
Imitation learning is a fusion between supervised learning and reinforcement learning:

**Question:** Given this relation what could be said about ways to solve IL?
The purpose of imitation learning is to efficiently learn a desired behavior by imitating an expert's behavior.

Can be achieved via:

1. Directly replicating the expert's behavior – 'behavioral cloning'
2. Learning hidden objectives of the expert's behavior – 'inverse RL'
3. Letting teacher interfere to correct bad behaviour – 'data aggregation'
4. Observing teacher solving an unfinished task – 'policy aggregation'
The purpose of imitation learning is to efficiently learn a desired behavior by imitating an expert's behavior.

Can be achieved via:

- directly replicating the expert’s behavior
The purpose of imitation learning is to efficiently learn a desired behavior by imitating an expert’s behavior.

Can be achieved via:

- directly replicating the expert’s behavior – ‘behavioral cloning’ (reduction to SP)
- learning hidden objectives of the expert’s behavior – ‘inverse RL’ (reduction to RL)
- letting teacher interfere to correct bad behaviour – ‘data aggregation’ (better reduction to SP+online learning)
- observing teacher solving an unfinished task – ‘policy aggregation’ (better reduction to SP+online learning)
The purpose of imitation learning is to efficiently learn a desired behavior by imitating an expert’s behavior. Can be achieved via:

- directly replicating the expert’s behavior – ‘behavioral cloning’ (reduction to SP)
- learning hidden objectives of the expert’s behavior
The purpose of imitation learning is to efficiently learn a desired behavior by imitating an expert's behavior.

Can be achieved via:

- directly replicating the expert’s behavior – ‘behavioral cloning’ (reduction to SP)
- learning hidden objectives of the expert’s behavior – ‘inverse RL’ (reduction to RL)
The purpose of imitation learning is to efficiently learn a desired behavior by imitating an expert’s behavior.

Can be achieved via:

- directly replicating the expert’s behavior – ‘behavioral cloning’ (reduction to SP)
- learning hidden objectives of the expert’s behavior – ‘inverse RL’ (reduction to RL)
- letting teacher interfere to correct bad behaviour – ‘data aggregation’ (better reduction to SP+online learning)
- observing teacher solving an unfinished task – ‘policy aggregation’ (better reduction to SP+online learning)
Key differences to RL

- not using rewards (at least not relying on them directly)
- use of examples or querying an interactive teacher
- structural constraints
- distribution shift
- cost of gathering examples
- presentation
- projects