Chapter 1

General MARL/PPO

1.1 The Surprising Effectiveness of MAPPO in Cooperative, Multi-Agent Games

https://www.researchgate.net/publication/349727671_The_Surprising_Effectiveness_of_MAPPO_in_Cooperative_Multi-Agent_Games

PPO sample efficiency

1 GPU desktop, 1 Multicore CPU for training

centralized value function - global state s insted of local o

environments: Particle world environment

PPO used be seen as sample less efficient, hence for academic purposes MADDPG anv value-decomposed Q-learning

Minimal hyperparameter tuning and no domain specification

Decentralized learning each agent its own policy, suffer from non-stationary transitions

two lines of research - CTDE (this) and value decomposition

in single agent PG advantage normalization is crucial

considered implementation details - input norm, value clipping, orthogonal init, gradient clip - all helpful and included

another - discretization action space for PPO to avoid bad local minima in continuous, layer normalization

MLP vs Recurrent

5 implementation details:
Value norm: running average over value estimates, value network regress to normalized target values (Pop art technique)
Agent-specific global state: concate of all o_i as input to critic
(agent specific global cannot be used in QMix, which uses single mixer network common to all agents)
Training Data Usage: importance sampling to perform off-policy correction ??
multiple epochs may suffer from non stationarity -¿ 15 to 5 epochs (easy to hard)
No mini-batches -¿ more data to estimate gradients -¿ imporved practical performance
Action masking: unavailable actions when computing action probabilities - both forward and backward
Death masking: zero states with agent ID as value input for dead agents

Chapter 2

Overcooked related

https://github.com/HumanCompatibleAI/overcooked_ai

2.1 On the Utility of Learning about Humans for Human-AI Coordination

https://arxiv.org/abs/1910.05789

agents assume their partner to be optimal or similar fail to be understood by humans

gains come from having agent adapt to human's gameplay

effective way to tackle two-player games is train agent with set of other AI agents, often past versions

collaboration is fundamentally different from competition (we need to go beyond self-play to account for human behavior)

incorporating human data or models into training leads to significant improvements (behavior cloning model)

Population Based Training is online evolutionary alg, adapts training hyperparameters and perform model selection agents, whose policies are parametrized by NN and trained with DRL alg. each PBT iteration pair of agents are drawn, trained for number of steps and have performance recorded at end of PBT iteration, worst performing agents are replaced with copies of best and parameters mutated

human behavior cloning peformed better than with Generative Aversarial Imitation Learning (GAIL)

PBT better than PPO self-play because they are trained to be good with population coordination

Agents designed for humans. Start with one learned human BC as part of environment dynamic and train single agent PPO.

start with ppo self-play and continue with training with human model

planning alg A*

two human behavior cloning models Hproxy used for evaluation and PPOBC learned against learned human models

best result self-play with self-play

for human interaction was best PPOBC with HProxy...PPOBC is overall preferable

PPOBC outperformes human-human performance

SP agents became very specialized and so suffered from distributional shift when paired with human models

future work - better human models, biasing population based training towards humans

READ AGAIN if interested

2.2 PantheonRL: A MARL Library for Dynamic Training Interactions

`https://iliad.stanford.edu/pdfs/publications/sarkar2022pantheonrl.pdf`

PantheonRL new software package for marl dynamic

Combination of PettingZoo and RLLIB - customziation of agents

prioritizes modularity of agent objects - each has separate replay buffer, own learning alg, role

(other powerfull DRL library - StableBaselines3)

The modularity of the agent policies com- bined with the inheritance of StableBaselines3 capabilities together give users a flexible and powerful library for ex- perimenting with complex multiagent interactions

2.3 Investigating Partner Diversification Methods in Cooperative Multi-agent Deep Reinforcement Learning

https://www.rujikorn.com/files/papers/diversity_ICONIP2020.pdf

PBT have diversity problem -¿ PBT agents are not more robust than self-play agents and aren't better with humans

creating diversity by generating pre-trained partners is simple but effective

(partner sampling - playing with uniformly sampled past versions of partner - lacks diversity, past versions have similar behavior)

(population-base training, pre-trained partners)

testing self-play and cross-play with these agent types (SP, SPpast, PBT, PTseeds, PTdiverse)

PTdiverse(hyperparameters) and PTseeds come from self-play

ref: ustesen, N., Torrado, R.R., Bontrager, P., Khalifa, A., Togelius, J., Risi, S.: Illu- minating generalization in deep reinforcement learning through procedural level generation. arXiv preprint arXiv:1806.10729 (2018)

IDEA to do: combine pretrained agents with maximum cross entropy? Encorporate maxium cross entropy into ppo??

LOVED THIS ARTICLE for it's simplicity

2.4 Evaluating the Robustness of Collaborative Agents

https://arxiv.org/abs/2101.05507

how test robustnes if cannot rely on validation reward metric

unit testing (from software engineering) - edge cases, eg. where soup was cooked but not delivered

incorporating Theory of mind to human model

human modal diversity by using population of human models

state diversity - init from states visited in human-human gameplay

test suite provides significant insight into robustness that is not correlated with average validation reward

"improved robustness as measured by test suite, but decrease in average validation reward"

simple ML metrics are insufficient to capture performance and we must evaluate results base on human judgement

domain randomization - some aspects of env are randomized - behavior can vary significantly based on small randomization of "irrelevant" factors

Theory of mind??? - each step agent decides what task/strategy to do (eg. deliver soup), then choose low-lever action (motion) to persue this goal.

Population of BC models, ToM models, or mixture of two

recurrent networks for all deep rl training procedures

"once the trained policy has found a good strategy for getting reward, it is not incentivized to explore other areas of the state space"

sampling initial state from human-human data (diverse starts)

Future work - how evaluate robustness in cases of ambiguous behavior

Or evaluating of proposals that populations with BC had positive effect, but negative for ToM

Meta learning for any kind of game layout, not just those prefabricated (online learning)

2.5 Interaction Flexibility in Artificial Agents Teaming with Humans

`https://www.researchgate.net/publication/351533529_Interaction_Flexibility_in_Artificial_Agents_Teaming_with_Humans`

too psychological, empirical studies of real people experience when playing with self-play / human BC agents

2.6 Maximum Entropy Population-Based Training for Zero-Shot Human-AI Coordination

`https://arxiv.org/abs/2112.11701`

"problem of training a Reinforcement Learning (RL) agent that is collaborative with humans without using any human data"

"To mitigate this distributional shift, we propose Maximum Entropy Population-based training (MEP). In MEP, agents in the population are trained with our derived Population Entropy bonus to promote both pairwise diversity between agents and individual diversity of agents themselves, and a common best agent is"

Comparing MEP with PPO self-play, PBT, Trajectory diversity and DIctitious CO-play

diverse and distringuishable behaviors between all agent pairs utilizes average KL divergence between all agent poairs

each agent in population is rewarded to maximize centralized population entropy.

we train best response agent by pairing it with with agent sampled by difficulty to collaborate with (prioritizing)

each agent has maximum entropy bonus (to reward) to encourage policy itself to be exploratory

Popuplation diversity = average entropy of each agent + average KL divergence of pairs

Populatino entropy = bounded and efficient surrogate for optimization = entropy of mean policies of population

PE is lower bound for PD

Not using PPO, but custom Entropy loss functions

Population entropy (effective linear pairwise kl divergence) as part of objective

MEP shares intuition with domain randomization... (MEP can be seen as domain randomization technique over partners policies)

(TrajeDi = encourages trajectories diffference between agents - Jensen-SHannon divergence between policies)

(Diversity-Inducing POlicy Gradient = formulated for single agent setting)

Bridges maximum entropy RL and PBT... entropy maximization for achieving robustness

Combining MEP with other MARL algorithms could be Future work

Idea: apply r=r+alpha * population entropy?

Idea: ("Maximum entropy approach adds the dense entropy to the reward for each time step, while entropy regularization adds the mean entropy to the surrogate objective") "Note that Entropy regularization is not, in general, equivalent to the maximum entropy objective, which not only optimizes for a policy with maximum entropy, but also optimizes the policy itself to visit states where it has high entropy. Put another way, the maximum entropy objective optimizes the expectation of the entropy with respect to the policy's state distribution, while entropy regularization only optimizes the policy entropy at the states that are visited, without actually trying to modify the policy itself to visit high- entropy state" `https://garage.readthedocs.io/en/latest/user/algo_ppo.html`

INTERESTING article

2.7 Assisting Unknown Teammates in Unknown Tasks: Ad Hoc Teamwork under Partial Observability

`https://arxiv.org/abs/2201.03538`

Ad hoc teamwork under partial observability (ATPO)

unknown teammates performing unknown task withou pre-coordination protocol

ad hoc teamwork has three parts - task identification, teammate identification and planning

What is Zero-shot coordination?? - studies how independently trained agents may interact with another on first-attempt

2.8

2.9 Work progress

7.11.2022

manipulace set agent id pro jednotliva env, jakym zpusobem se zpracovavaji odesilane obs a prijimane actions je potreba toto pak jakkoliv resit na

urovni stable baselines strev? vypada to, ze ne ... Na strane RUNNER: v *obs*[0] je vzdy obs pro self.model a v *obs*[1] je vzdy obs pro self.other agent model Runner vzdy vytvori joint actions jako (self actions, other agent actions) a posle je do env Env je zpracuje, podle agent id budto necha, nebo spravne prohodi do joint actions Env s joint actions provede step, ziska obs a opet spravne dle agent idx budto necha nebo prohodi a vraci (obs0, obs1), A opet jsme na zacatku ... Runner se muze spolehnout ze v *obs*[0] ma obs toho trenovaneho modelu self.policy a v *obs*[1] ma obs pro other agent model (de facto embedded into environment)

kolik env steps provadi Runner? nelze nastavit pevne jedna epizoda == 400? =¿ parametr v PPO()

proc overcooked env reset musi resetovat mlam? Zatim zakomentovano, TODO: proverit

learning rate zatim neni annealovany

zakomentovany nektere metriky, kterym zatim nerozumim nebo se mi nemeni (clip fraction, clip range, learning rate)

15.11.2022

Struktura inspiravana projektem max population entropy, take pouzito stable baselines

stable baselines posledni oficialni podpora TF1, neoficialni podpora TF2

nove doporucene Stable baselines3 ktere pouziva pytorch

S pytorch moc neumim, ale pozmenit reward a pridat do loss rozdil oproti populaci asi neni problem, takze zalezi jak moc velke zmeny ocekavam

Adaptace na SB3 docela jednoducha, vektorizovane prostredi

Other agent jako soucast "single agent" prostredi

Technicke problemy:

Prostredi se vzdy resetuje do stejneho stavu

Struktura CNN, MLP vs reprezentace stavu lossless, featurize state mdp

RNN, Frame stacking, nebo staci reakcni agent?

Jak vyhodnocovat agenty deterministicky arg max dopadne kazda epizoda naprosto stejne, nedeterministicky teoreticky nedostavam nejlepsi vysledek, nebo vyhodnocovany agent deterministicky a other agent nedeterministicky? Zatim nevim jak to resi v ostatnich projektech

Annealing entropy coeficient?, annealing sparse r coef

Napevno 5M env steps staci? Da se o tom rict "with little loss of generality"? Nebo tohle muze byt zkreslujici pro vysledek mych experimentu?

5 zajimavych map

diff bonus: k dispozici log_prob pro danou akci, $e^log_prob = prob, diff = ?min_pop(p(a) - a(a))^2 = ?(mean(pop(a) - a(a)))^2$

Konkretni plan: Zacit s jednou mapou, natrenovat si 10 self-play agentu (pripadne dalsich 10 s ruznymi hyper-parameters), ktere si necham pro nezavisle testovani, Pak zacit trenovat tim pridavacim zpusobem populaci a kazdeho noveho jedince otestovat vuci vsem 10 testovacim. Teorie je takova ze s kazdym pozdeji pridanym agentem ma byt vysledek lepsi vuci test agentum. Zacit s diff bonusem = 0 pro porovnani a pak zkouset bonus zvysovat (0.1). To same pro dalsi mapu a zkusit najit nejakou zajimavou hodnotu pro diff

2.12.2022

Zatim se nedari dosahnout SP off-diagonal failure

Zkousim vymyslet stejnou hodnotu entropy coef napric vsemi expermienty. Pro 0.01 parkrat nezkonvergovalo. Zkousim, jestli nepomuze prodlouzit dobu ziskavani castecnych odmen.

Zkousim jak moc by slo snizit celkovy pocet kroku trenovani, aby to bylo jeste stabilni a zaroven aby to netrvalo tak dlouho

Chce to zrefaktorovat a rozmyslet strukturu evaluace/treninkovych zpusobu a mnozin/ vizualizace, aby vse fungovalo obecne

Z dnesni spolecne konzultace s Martinem a Petou -¿ Peta se planuje zabyvat komplexnejsimi mapami nebo komplexnejsim prostredim (pr. vice receptu) Zatim se mi nedari moc nasimulovat ten problem s robustnosti, velka cast agentu se mi zatim jevi jako relativne kompatibilni. Komplexnejsi prostredi by to treba mohlo pomoct rozbit. Zaroven by tam pak chybelo porovnani vuci jinym jiz existujicim pracim, protoze vsechny prace docela shodne pracuji jen s pevnou mnozinou 5 pevnych map

TODO: Potreba rozmyslet metriku pro rozdil vuci populaci, MSE se mi zda ze je stejne jako rozdil vuci prumerne distribuce populace, coz mi Martin rozmlouval Mozna nejakou prob dist. metriku, KLL divergenci?

4.12.2022

Mozna souvislost mezi off-diagonal faileru a entropy coeficient Za pozornost stoji SP_RS_E0_IMPORTANT_Entropy.png, kde prvni polovina agentu je natrenovana s ent coef == 0 zatimco druha polovina s ent coef == 0.02 Provnanim leveho horniho ctverce a praveho dolniho ctverce vidime velky rozdil ve vysledcich

Zkousim ted rychly experiment (1M stepů) s coef 0, 0.01, 0.02 a uvidime

Pro nizke ent coef se casto stava ze chovani nezkonverguje k necemu rozumnemu - pridal jsem early stopping kdyz agent nezvladne v prvnich N krocich uvarit jedinou polevku

prechozi verze evaluatoru vyuzivajici MDP evaluator se mi zdala ze fungovala zvlastne (pr. SP agenti obcas dopadli se sebou samymi katastrofalne) Implementoval jsem evaluator, ktery se chova co nejvic podobne simulacim pri trenovani, jen vzdy preferuji deterministicke akce Pripada mi, ze vypada lepe - diagonala neselhava pro SP TODO: zjistit proc se lisi od originalniho evaluatoru... Neni tam chyba?

TODO: refaktoring struktury a nazvu porovnani dvou mnozin

TODO: device cuda funguje daleko pomaleji nez cpu Nedari se mi pouzivat cpu pro predikce pro training samples a cuda pro samotny learn

6.12.2022

Implementoval jsem early stopping evaluaci, kdy koncim SP trenink, kdyz evaluace konci nad threshold hodnotou Tim padem minimalizuju riziko ze na diagonale budu mit spatne hodnoty

Pripada mi, ze s early stopping nemusim nutne resit nejakou pevnou hodnotu entropy coeficientu. Snad by mohlo stacit zacit s vysokou entropii, annealovat ji k nule a hledat early stopping reseni

Kdyz jsem pouzil pri treninku vic random stavy (p_rnd v start_state_fn), nez pak pri evaluaci, tak to vypadalo ze se zlepsila situace na diagonale Tim padem se by se snad mohla "zhorsit" off diagonal

Naopak pri mene random trenink stavu nez pri evaluaci, tak to vedlo k vic random vysledkum

TODO: Zrejme ma smysl pri treninku vzit co nejvic random stavy. Ma smysl pri evaluaci random snizit?

TODO: Stale neni vyresene jak vyhodnocovat. Napady pro SP vyhodnoceni: Pevny threshold ¿ True or False a spocitat pocet Jako threshold pouzit na kazde pozici hodnotu z diagonaly Jak se postavit k porovnani mych vysledku vs vysledkum v clanku?

Napady pro vyhodnoceni populace (VS SP mnozina): Prumer jedince z populace vuci vsem SP agentum Zkouset zase pocet nad nejaky threshold?

TODO: Nejake napady, jak prostredi ztizit? Jestli se tim nedosahne vetsi off-diagonal SP failure. Co pak s porovnatelnosti vuci jinym clankum?

Konzultace 6.12.2022

Mozna se ten off-diagonal problem neprojevuje kvuli tomu, ze pracuji s jednodussim prostredim Tj. nevyuzivam loss_less state a tim padem ani konvolucni site

Martin si mysli jestli kvuli tomu jinemu nastaveni prostredi nekonverguji vsechny agenti ke stejnemu/podobnemu chovani

Mozne dalsi kroky: Vyzkouset, jestli bude situace stejna na ostatnich mapach. Mozna jsou ostatni tezsi.

Vyzkouset zmenit na loss_less a pridat konvolucni vrstvy viz (https://arxiv.org/abs/1910.05789v2)

Konzultace 13.1.2023

Loss_less funguje, chyba v normalizaci

hyperparametry dle clanku moc nefunguje

nalezl jsem parametry s vahou critic aspon 0.1 ktere mi funguji pro vsechny mapy... random zkousenim cca 100 moznych kombinaci hyperparametru

Obcas selze vypada ze pouze pro posledni mapu - mam predcasne ukonceni po 2.5M (cca 12 minut), mozna by mohla byt situace jina s "chytrymi pomocniky"

cramped rooom a asynchronous advantages moc nemaji off-diagonal problem, plus vetsi randomizace prostredi to jeste zhorsuje, nebo stezuje uceni

Pripada mi ze entropy coef 0.1 byl moc silny, ale pouze domnenka, nemam cas se tomu vic venovat

Plan pokracovat na trech mapach Pro kazdou predtrenovat 3x10 vyhodnocovaci skupiny

Zacit s populacnim trenovanim - dve myslenky, pridat inverse annealing spolecne reward

Obcas ma vysledny model na konci eval 0 prestoze v prubehu na tom nebyl spatne - tj. budu asi brat nejlepsi model z prubehu. V pripade populace rozmyslet eval, vuci vsem v populaci, aby to nebylo zkreslene

Zacit s pop diff rewardem, MSE na prop vectoru asi neni dobry napad, tak spis KL, po konzultaci neresim poradi vstupu do KL_DIV, proste jedno vybereme

2.9.1 Summary 25.1.2023

Predelal jsem args na Namespace argparses, aby slo dobre volat z prikazove radky

Prosel jsem si nezavisle conda instalaci bez instalace samotneho stable-baselines3 a overcooked, protoze existuji jako clone v projektu. Momentalne zjistuju jak vytvorit shell script ktery zvladne aktivat conda env a pustit entrypoint Zbyva asi vyresit sys.path.append cesty k overcooked_ai_py a stable_baselines3

TODO: zkontrolovat co se vlastne pocita v tom mem evaluate. Pripada mi ze to nevraci hodnoty c*20 / #parallel envs

Pro populacni trenovani jsem opustil myslenku jednoho pevneho soupere pro celou trenovaci epochu. Namisto toho rozdelim populaci mezi 30 paralelnich prostredi vuci kterym trenuju. Tim mam zajisteno, ze to pri treninku stale vidi celou populaci. Tento pristup hodne zpomaluje trenovani, protoze najednou musim v kazdem stepu volat vickrat NN. Pro vyreseni nedelitelnosti na zacatku treninkove epochy populaci shuffluju

Zkousel jsem pri populaci dva ruzne pristupy vzhledem k shared rewards. Prvni je nechat shared rewards tak jak jsou, toto mi nepripada intuitivni, nebot muze stacit aby partner vedet jak varit polevku a pak se ani nemusim snazit/nemusim na tom mit podil a presto mam odmenu. Proto jdu druhou cestu, kdy inverse annealuju shared rewards opacne k tem partial rewards. Tedy intuice je takova ze nejprve se ucim jak se delaji subtasky a pak az je budu umet, tak se budu vic zabyvat tim jak by bylo nejlepsi abych to delal s kuchary v populaci.

KL div reward vuci populaci zatim beru jako minimum ze vsech kl div. TODO: rozmyslet jestli by nebylo lepsi pocitat prumer pres vsechny. Je potreba taky se zamyslet nad koeficientem tohoto rewardu a asi nejspise i jeho clipingem. Tj. aby byl dost silny na to aby mel dopad, ale zaroven hlidat jeho meze, aby se stale resil puvodni problem.

Rozmyslet jako roli hraje KL div vuci entropii. Jestli v KL Div uz neni entropie zachycena. Proto nejspis pak pro modifikaci loss nechat pouze cross entropii.

S kazdym dalsim jedincem v populaci prodlujuzu trenink dalsiho o 1e5 kroku

Pro kazdou mapu jsem predtrenoval 30 SP agentu, ktere planuju vyuzit jako eval mnozinu.

Prvni jedinec v populacnim treninku je SP, otazka jestli nepouzivat pro jednu mapu vzdy toho stejneho, aby se v experimentech zacinalo ze stejne strategie. Pak by se mozna lepe porovnavalo jak moc se dokazala populace zdiverzifikovat.

TODO: should population partners play argmax during training?