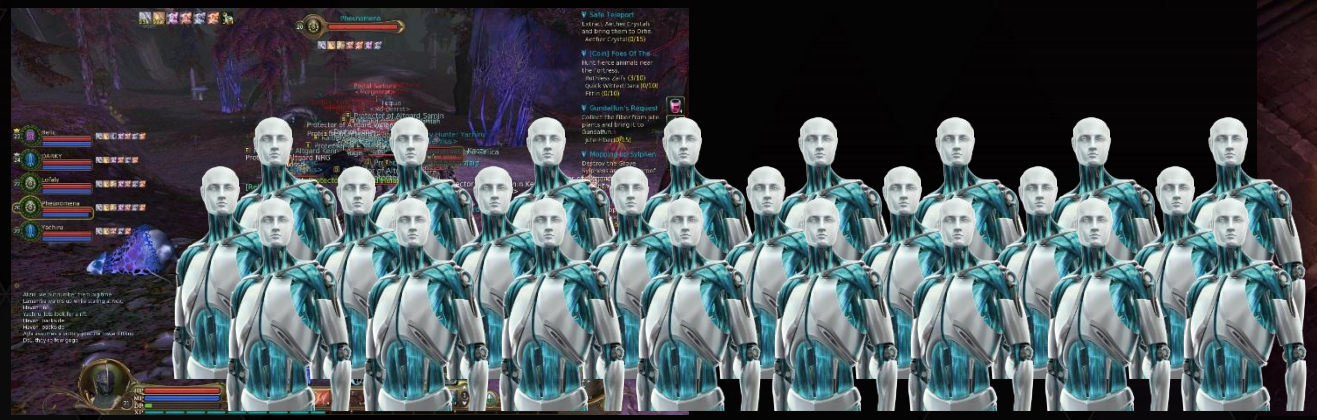
GPU TECHNOLOGY CONFERENCE

MULTI AGENT GAME AI DEEP LEARNING CLUSTER

성낙호, NCSOFT

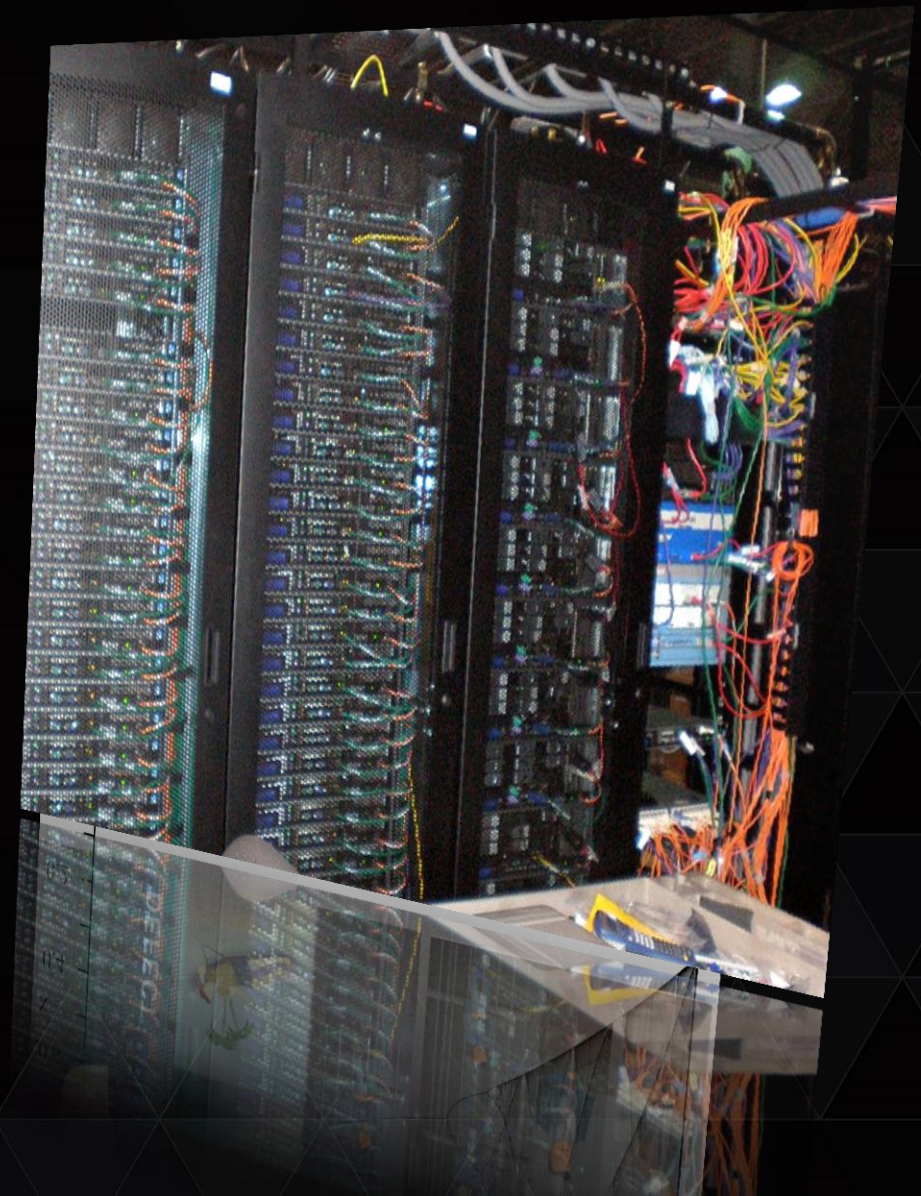
MISSION

- Diablo, AION, LOL 같은…
- Multiplayer RPG 게임에서의 파티 플레이가 가능한 전투 AI를
- 실제 서비스에 사용할 수준으로 개발



CHALLENGE

- 게임 규칙의 복잡도
- 멀티 플레이
- Real-time



어려움#1 – 복잡한 규칙

- 플레이어
  - 고유 캐릭터
- 캐릭터
  - 고유 스킬, 능력치, 상태값, 물리적인 '실체' (연속 좌표계, 2D/3D)
- 스킬
  - 고유 사용 조건, 효과
- 효과
  - 상태 이상을 포함한 다양한 능력치, 상태값 변경 규칙

- … '규칙성' 없는 규칙들의 집합

어려움#2 – 다자간 협력 플레이

‘오더’를 내리는 사령관 없이, ‘스스로’ ‘팀 플레이’

TOO COMPLEX TO EXPLORE/CODE

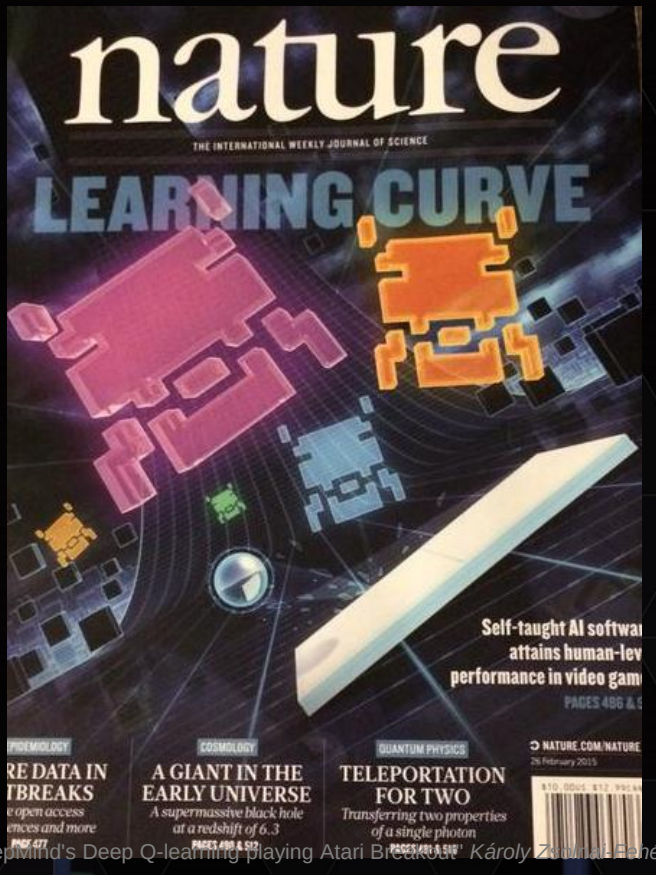
- Compute:
  - 기존의 AI algorithm을 적용하기에는 너무 복잡함
  - 실제 서비스 가능한 수준의 계산량을 크게 상회함
- Code:
  - Handcrafted algorithm으로 대처하기에는 감당할 수 없는 유지 보수 비용

REINFORCEMENT LEARNING

이전의 GAME AI 연구

- MCTS Mario – Non real time
- Deepmind – Simple atari game

MCTS Mario. "MCTS Super Mario AI #4 - Final Combination." *Emil Juul Jacobsen.* YouTube,



"Google DeepMind's Deep Q-learning playing Atari Breakout" *Károly Zsolnai-Fehér.* YouTube,

START SMALL GROW BIG

- 핵심 문제 속성을 공유하는 실험 게임 제작

Mini LOL

- 두 개의 클래스, 간단한 맵
- 그래픽 요소 제거한 텍스트 기반 게임

MINILOL: EFFECT

- DirectDamage
- Buff
- Debuff
- Stun
- Heal
- Cure
- HealPerTick
- MPHealPerTick
- Dash
- HealReflect
- SplashDamage

MINILOL: RENEKTON

양떼 도륙
C: 7, CT: 15, R: 2.4
DirectDamage / HealReflect

무자비한 포식자
C: 12, CT: 30, R: 2.4
DirectDamage / Stun

자르고 토막내기
C: 18, CT: 21, R: 3.0
DirectDamage / Dash

강신
C: 24, CT:90, R: 3.0
Heal / MPHealPerTick

http://leagueoflegends.wikia.com/wiki/Renekton

http://leagueoflegends.wikia.com/wiki/Kayle

MINILOL: KAYLE

징벌
C: 12, CT: 7, R: 3.8
DirectDamage / DeBuff (Speed)

신성한 축복
C: 27, CT: 45, R: 6.0
Heal / Buff (Speed)

정의로운 분노
C: 17, CT: 25, R: 1.0
Buff (Range)

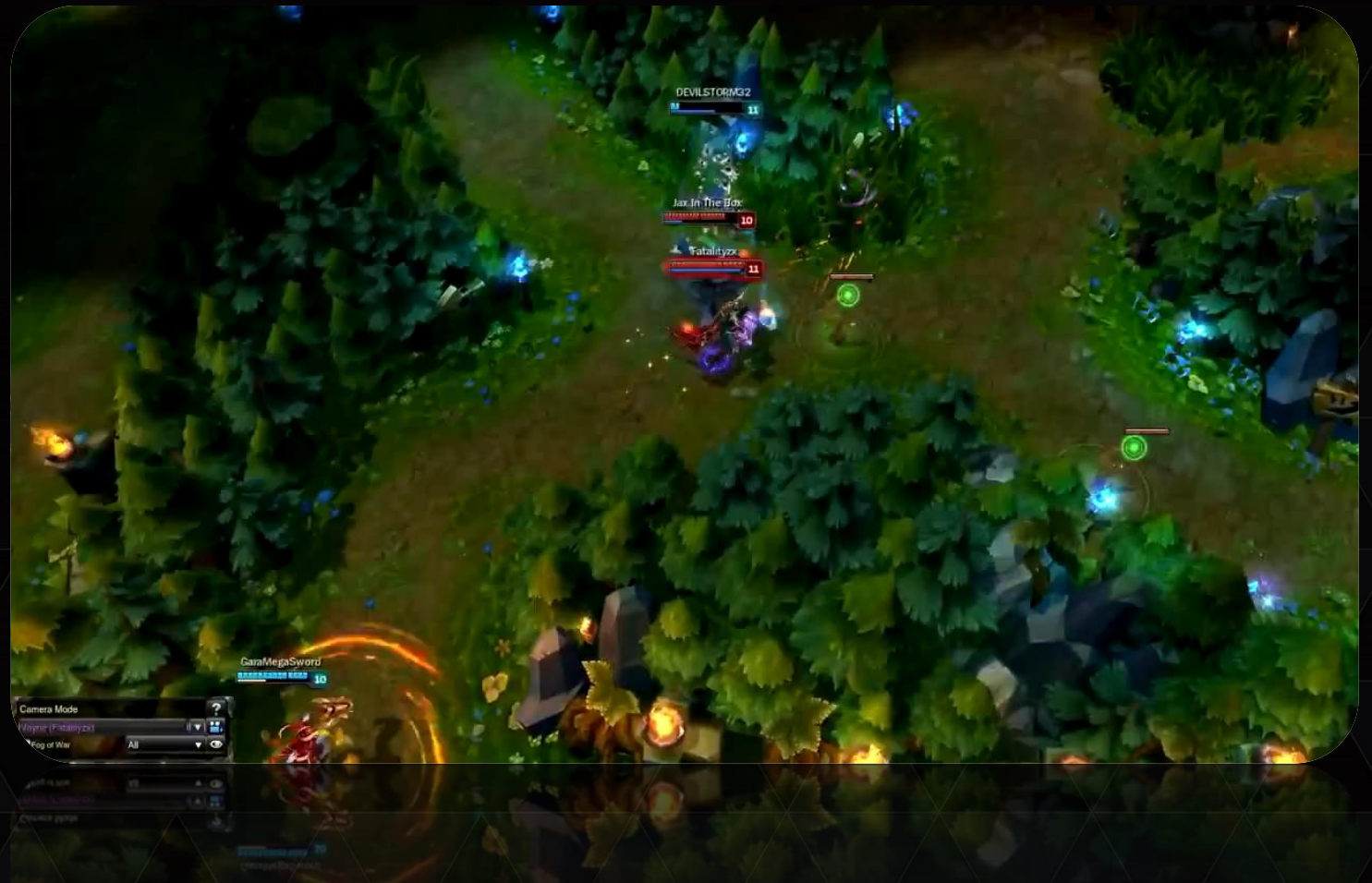
중재
C: 0, CT:90, R: 1.0
Buff (Invincible)

BLACKBOX LEARNING

게임의 규칙을 전혀 모르고 만든

Reinforcement learning agent

START SMALL TO GROW BIG



자동화된 AI 경연장

- 여러 종류의 방법론을 실험하기 위한 경연장
- 다양한 매치업 매치를 동시에 진행
  - Behavior tree
  - N vs N
  - Class
  - Map
    - Hollow, bush, wall,
- 새로운 에이전트의 능력을 객관적 지표화
  - Evaluation function
    - 1 on 1, 3 on 3, 5 on 5
    - simple greedy, expectation based greedy
  - Reinforcement learning
    - 1-class, 2-classes
    - Deep learning

기록: 2014.12

	1on1	5on5	5on5/fixed	bushes	hollow	wall/bush
DQN	156	136	132	133	141	139
greedy/best	133	107	109	113	112	118
BT/best	90	109	114	115	115	115

영상: 2014. 12

경연의 기록

- 2014.11
  - Simple greedy(evaluation function) agent의 우세
  - Behavior tree agent의 역습
  - 개선된 greedy agent의 완승
- 2014.12
  - Deep q learning agent의 역습 (4 layers, drop-out)
  - Expectation-based greedy agent의 복수
- **2015.3**
  - **CUDA cluster 구축**
  - **LSTM agent의 완승** (rmsprop, 7 layers, max-out)

GTX980 X 2 X 30

현재까지의 성과

- >90% 승률:
  - Hand crafted feature (거의) 없이

경쟁 상대 AI 개발 중단으로 연구 종료

Q LEARNING

$$Q : S \times A \rightarrow \mathbb{R}$$

$$Q_{t+1}(s_t, a_t) = \underbrace{Q_t(s_t, a_t)}_{\text{old value}} + \underbrace{\alpha_t(s_t, a_t)}_{\text{learning rate}} \cdot \Bigg(\overbrace{\underbrace{R_{t+1}}_{\text{reward}} + \underbrace{\gamma}_{\text{discount factor}} \underbrace{\max_a Q_t(s_{t+1}, a)}_{\text{estimate of optimal future value}}}^{\text{learned value}} - \underbrace{Q_t(s_t, a_t)}_{\text{old value}} \Bigg)$$

APPROX Q LEARNING

- 연속 상태 공간을 다루기 위해 필연적

- Neural fitted Q Iteration (Riedmiller, 2005)

DEEP Q LEARNING NETWORK

- Supervised learning
  - Input: SxA, Target: Q
- Iterative learning
  - Q value는 iterative algorithm으로 update됨

NFQ의 어려움

- 두 개의 반복 알고리즘

- 전체 성능이 안정적이지 않은 양상을 띔

QUEST: “Q-VALUE를 학습하세요”

- State, action, reward design
- 불안정한 학습
- Over-fitting
- Under-fitting

STATE, ACTION AND REWARD

- State := (global, self, friend-1, friend-2, .., enemy-1, enemy-2, …)
- invariance

STATE, ACTION AND REWARD

- Action := (Move-commands, Action-commands)

- Unrolled

  - Move-commands := (Stop, [MoveAction.Id, Target.Id]*)

  - Action-commands := ([Action.Id, Target.Id]*)

STATE, ACTION AND REWARD

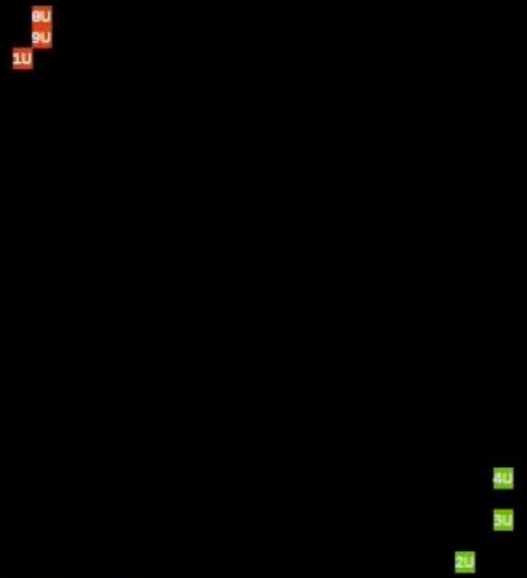
- Team reward
  - 팀의 승리를 위해서는 기꺼이 자신을 희생할 수 있도록

STABILIZERS

- Many simultaneous simulators
  - ~30 simulators
- Freezing network (Deepmind, 2015)
  - Network - double buffering

LEARNING TECHNIQUE

- Normalized feature
- rmsprop/DropOut for deeper network
- MaxOut
- Hyper parameter optimization
- LSTM



Status 1

Status 2

Status 3
Perfs(4, AIOnTick) consume sec : 0.000054

Standing
--victory status--
Team1(RemoteQL1) 18 : 5 Team2(EvalFnAI_v0.4_New)
T1 WinPerc: 78.26% (50-MA:78.26%)

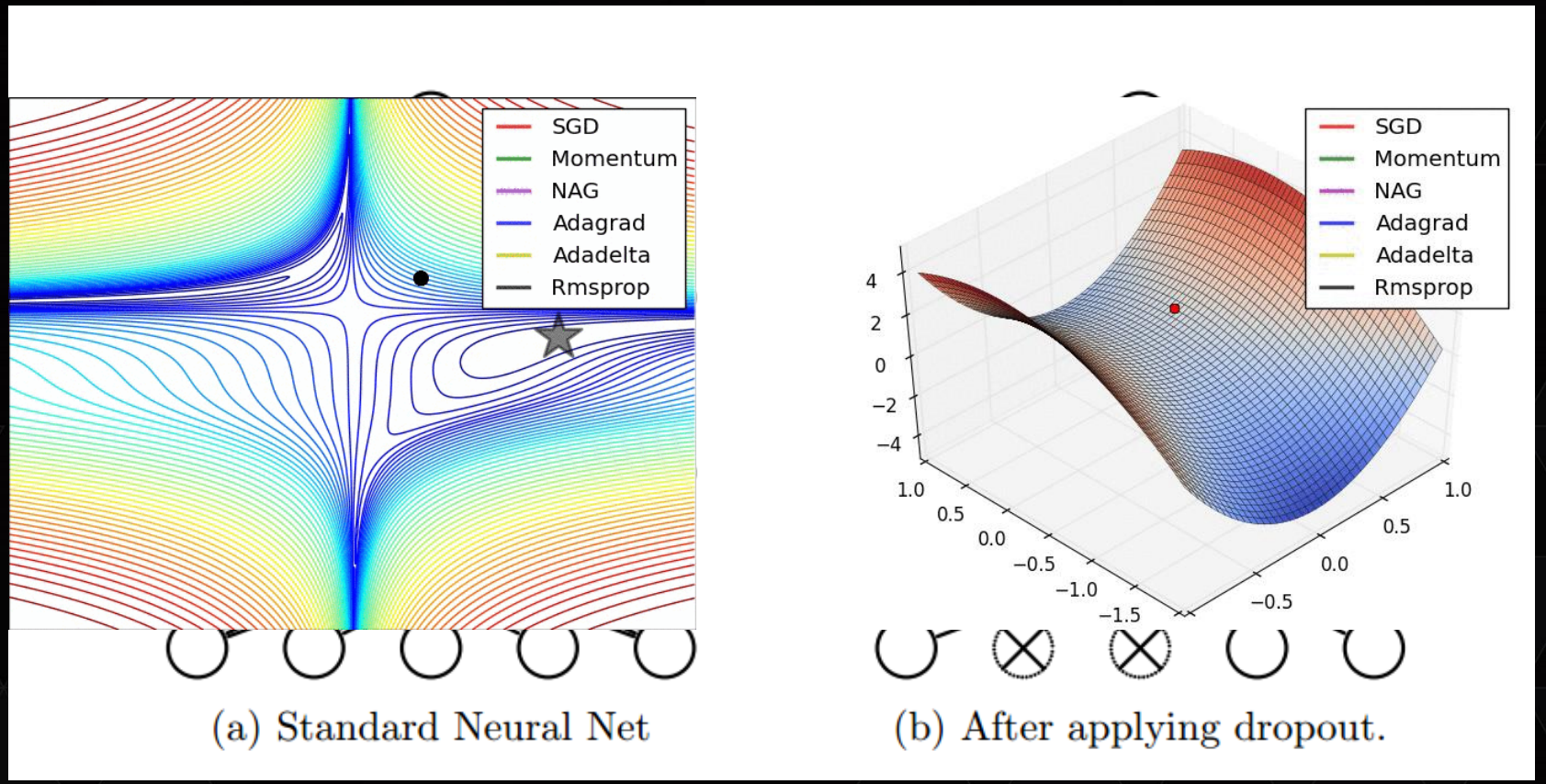
Events

NORMALIZED FEATURE

- 빠른 학습
  - Network의 빠른 학습을 위해 feature는 [-1,1] 혹은 [0,1]로 normalize
  - Ranged value의 경우, tile-mapping 적용
  - 참고: Batch normalization

- 예
  - Health : Health / MaxHealth → [0,1]
  - Distance : [0,최대공격거리), [최대공격거리,최대시야거리) tile-mapping

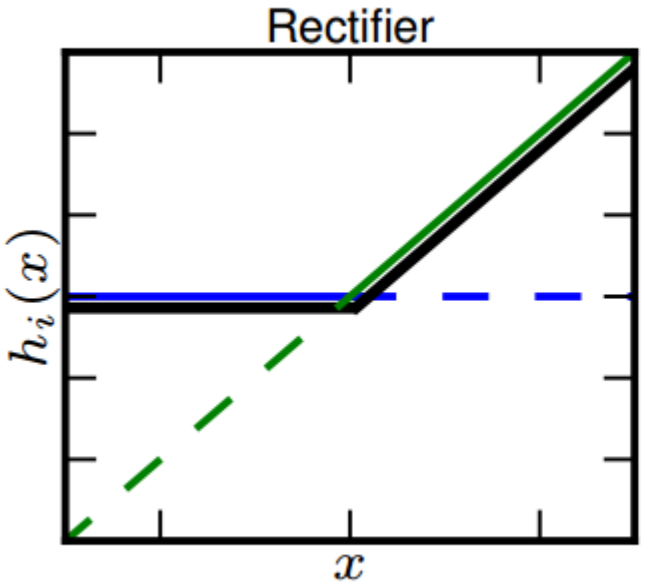
GOING DEEP: DROP OUT, RMSPROP

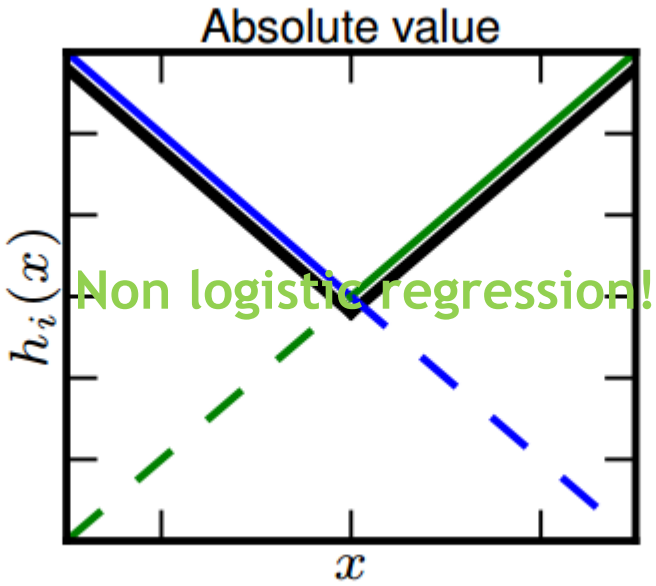


http://cs231n.github.io/neural-networks-3/

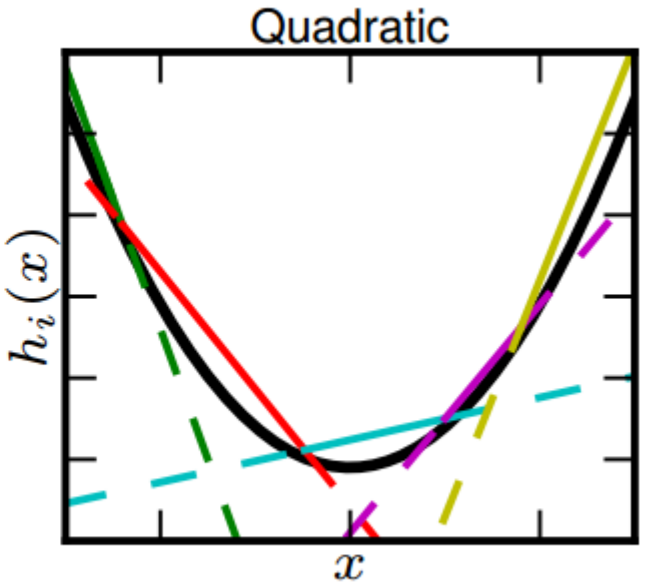
Srivastava, Nitish, et al. "Dropout: A simple way to prevent neural networks from overfitting." *The Journal of Machine Learning Research* 15.1 (2014): 1929-1958.

MAX OUT





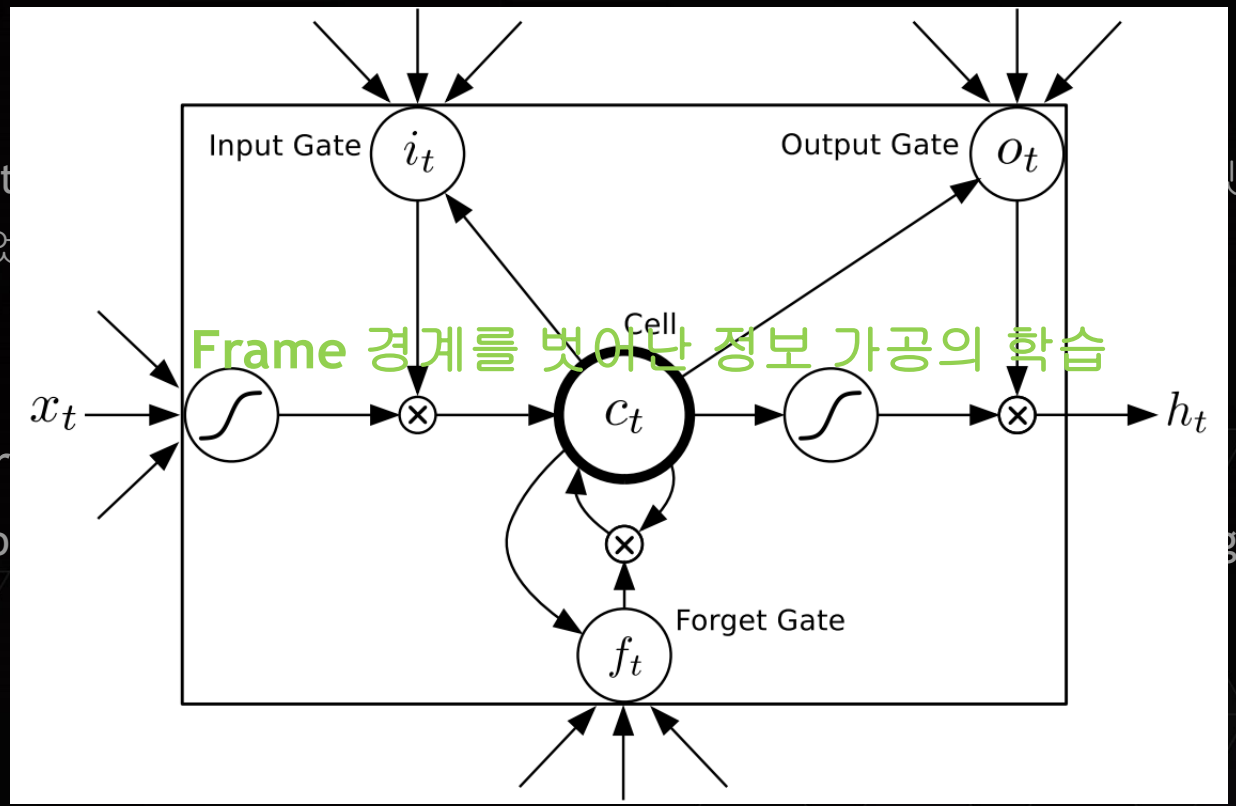
Non logistic regression!



Goodfellow, Ian J., et al. "Maxout networks." *arXiv preprint arXiv:1302.4389*(2013).

LSTM

- 고차원
  - Stat
  - 무엇
- Back pr
  - Rep



Frame 경계를 벗어난 정보 가공의 학습

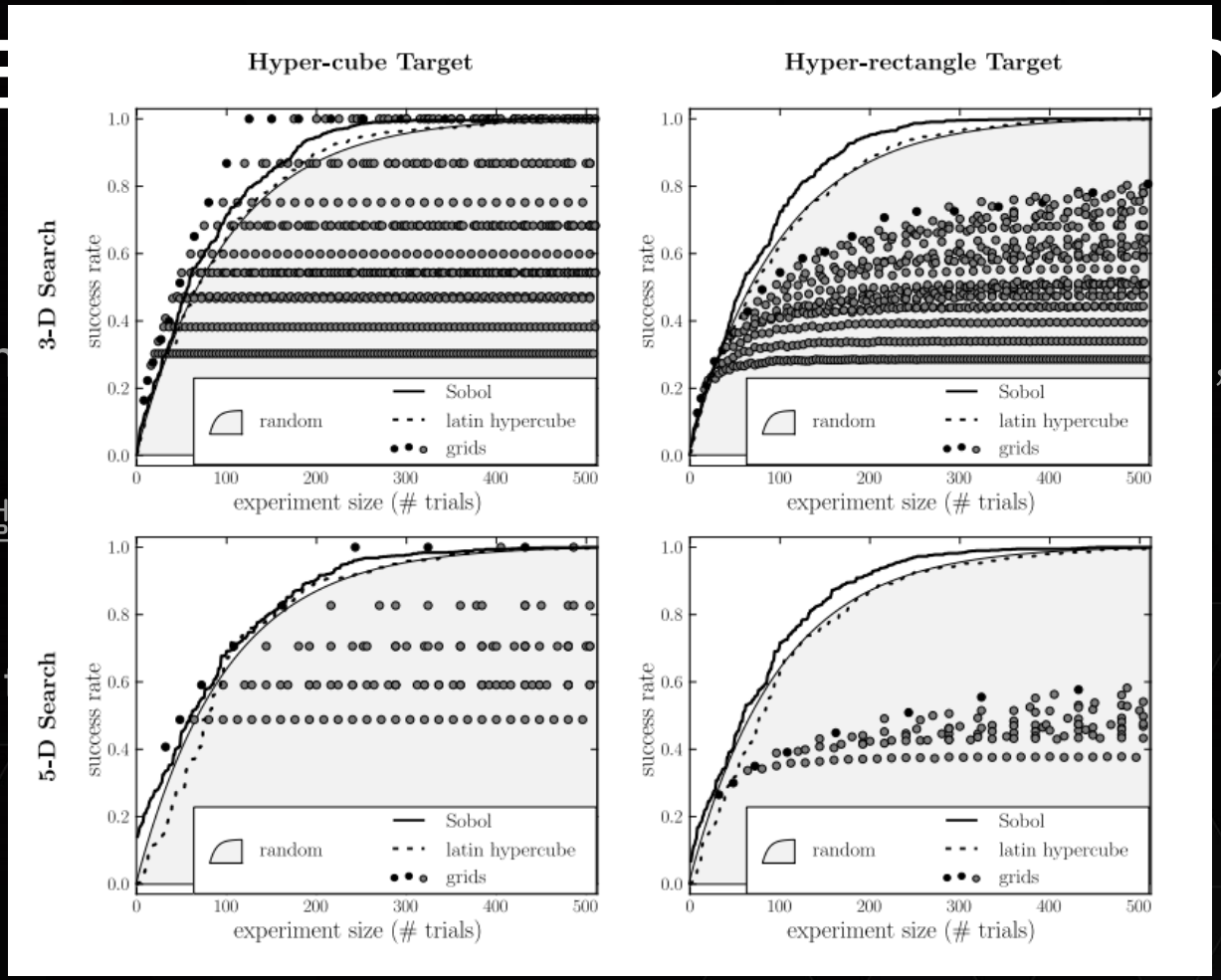
http://blog.otoro.net/2015/05/14/long-short-term-memory/

MEASURE AND TUNE

- 네트워크의 성능 ➔ Agent 성능 반복 측정
- 모델의 성능 ➔ Iteration별 네트워크 성능 추세
- 최적화된 hyper parameter ➔ 측정의 반복!

HYPE ON

- Rando , 2012)
- 미리 설
- Curren



Bergstra, James, and Yoshua Bengio. "Random search for hyper-parameter optimization." The Journal of Machine Learning Research 13.1 (2012): 281-305.

IN-HOUSE CLUSTER

- 시도해 봐야 할 많은 조합 ➔ **Data parallelism**

STACK

- Docker
- CUDA
- Caffe
- C++11
- Node.js (websocket, angular.js)
- 0-mq
- Redis

IN-HOUSE CLUSTER

- Dockerize everything
- 구현 비용 : ~2MM

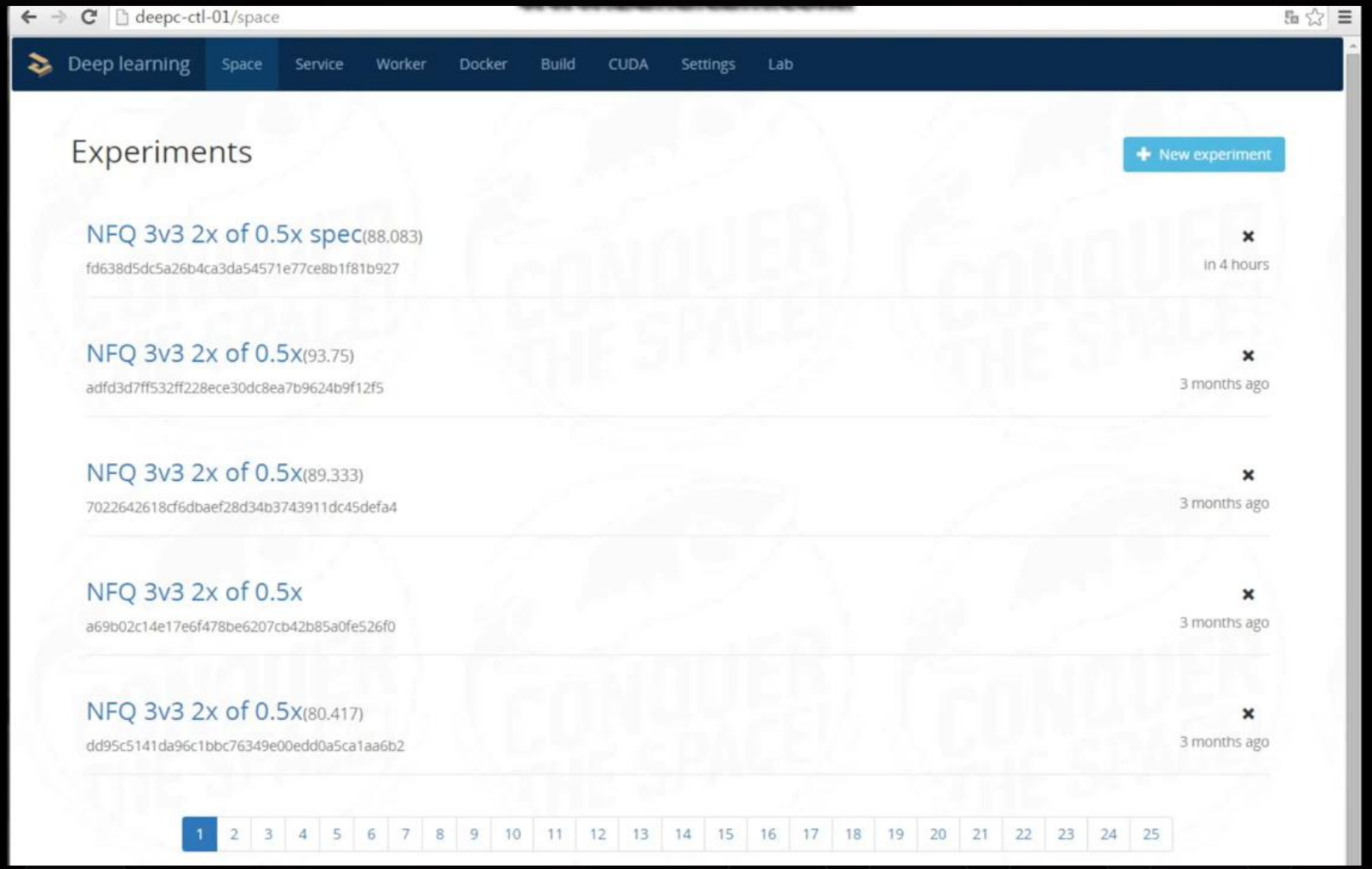
FASTER EXPERIMENT

- Automated build / deploy
  - 소스 코드가 gitlab에 push 되면,
  - Compute machine의 docker에 각각 자동 build 요청이 들어가고,
  - 새로운 version으로 자동으로 배포됨

<10s : 코드 변경 → 배포/실행

PLUGGABLE SIMULATOR

- Simulator 독립적인 구조
- 현재까지 구현된 simulator
  - MiniLOL
  - UnrealEngine4 Deep learning plugin



다음 목표

- 非-전투 게임 플레이 학습
- Offline learning
  - 서비스 중인 게임에서의 로그 기반으로 학습
- Low resolution convolution network
  - 지형 지물, war fog 등의 정보

GPU TECHNOLOGY CONFERENCE

감사합니다

JOIN THE CONVERSATION

#GTC15