AI & Automation at King

Sahar Asadi

We make great games

- We have developed more than 200 fun titles and our games can be played and enjoyed all over the world
- King had 258 million monthly active users for the quarter (Q2 2019)
- The company has been part of Activision Blizzard since February 2016

King has offices or studios in Stockholm, London, Barcelona, Malmo, Berlin, San Francisco, Chicago, New York, Los Angeles and Malta.

Some stats and facts

Global leader in cross-platform casual games

Four global franchises:

Candy Crush

Pet Rescue

Farm Heroes

Bubble Witch

Founded in 2003, studios in Stockholm, London, Barcelona, Malmo and Berlin.

Employees (approx.)

2000

AI R&D Team @ King

Analytics Strategy & Platform

ML Platform

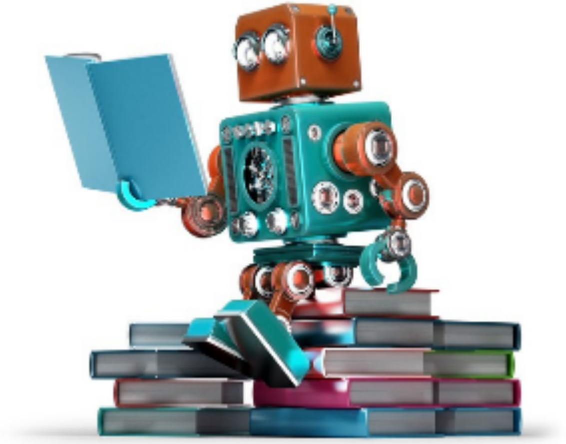
Exploratory Research

Product Use-cases

Research Areas

Personalization

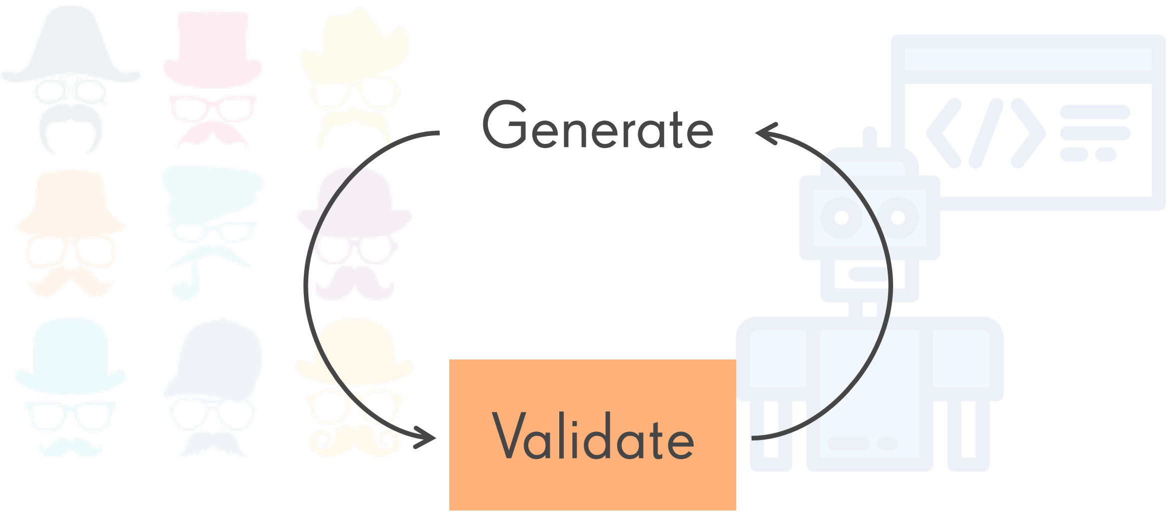
Development Automation



Content Generation

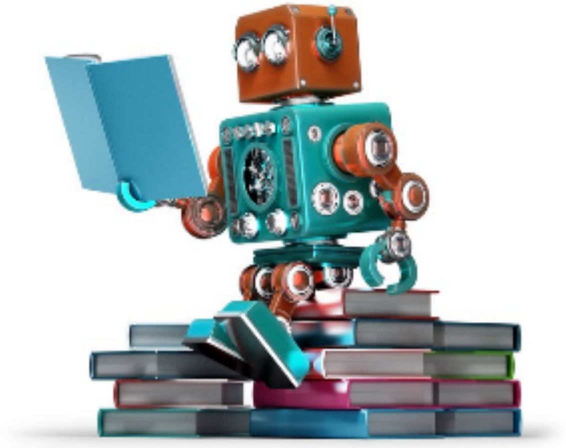
Picture in the middle: made by xnimrodx from https://www.flaticon.com/

Research Areas



Personalization

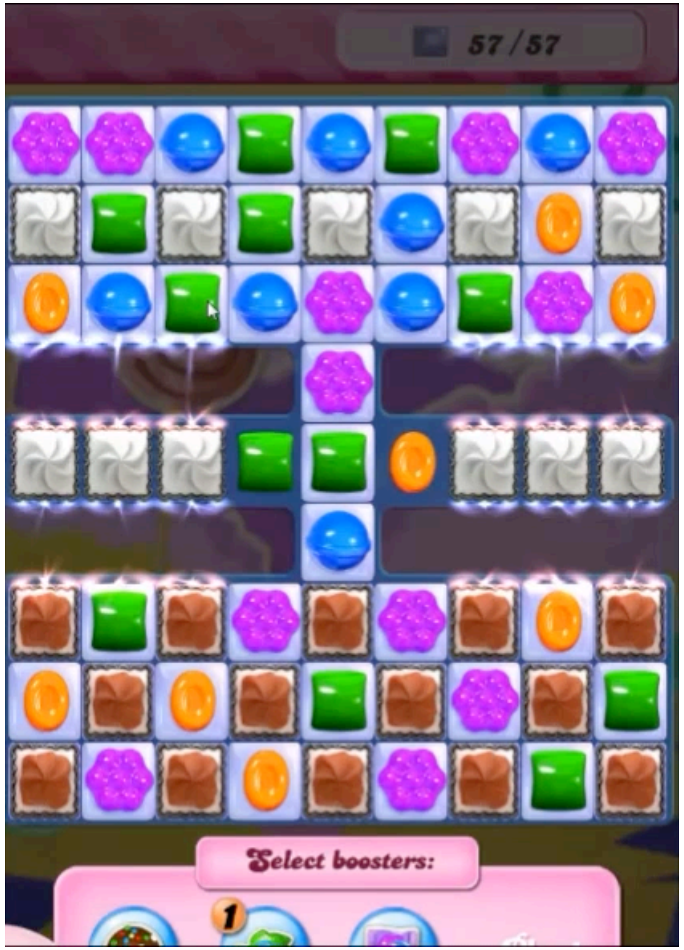
Development Automation



Content Generation

High Quality Content

What is content?



Strategy

Non-deterministic

Balanced

Aesthetics

Unique

Large state space

Content production pipeline

Creation	Balancing	Release
Create a new level	Modify and tweak	Staged release
Maximum creativity	Make enjoyable for everyone	Accessible to players

Content production pipeline

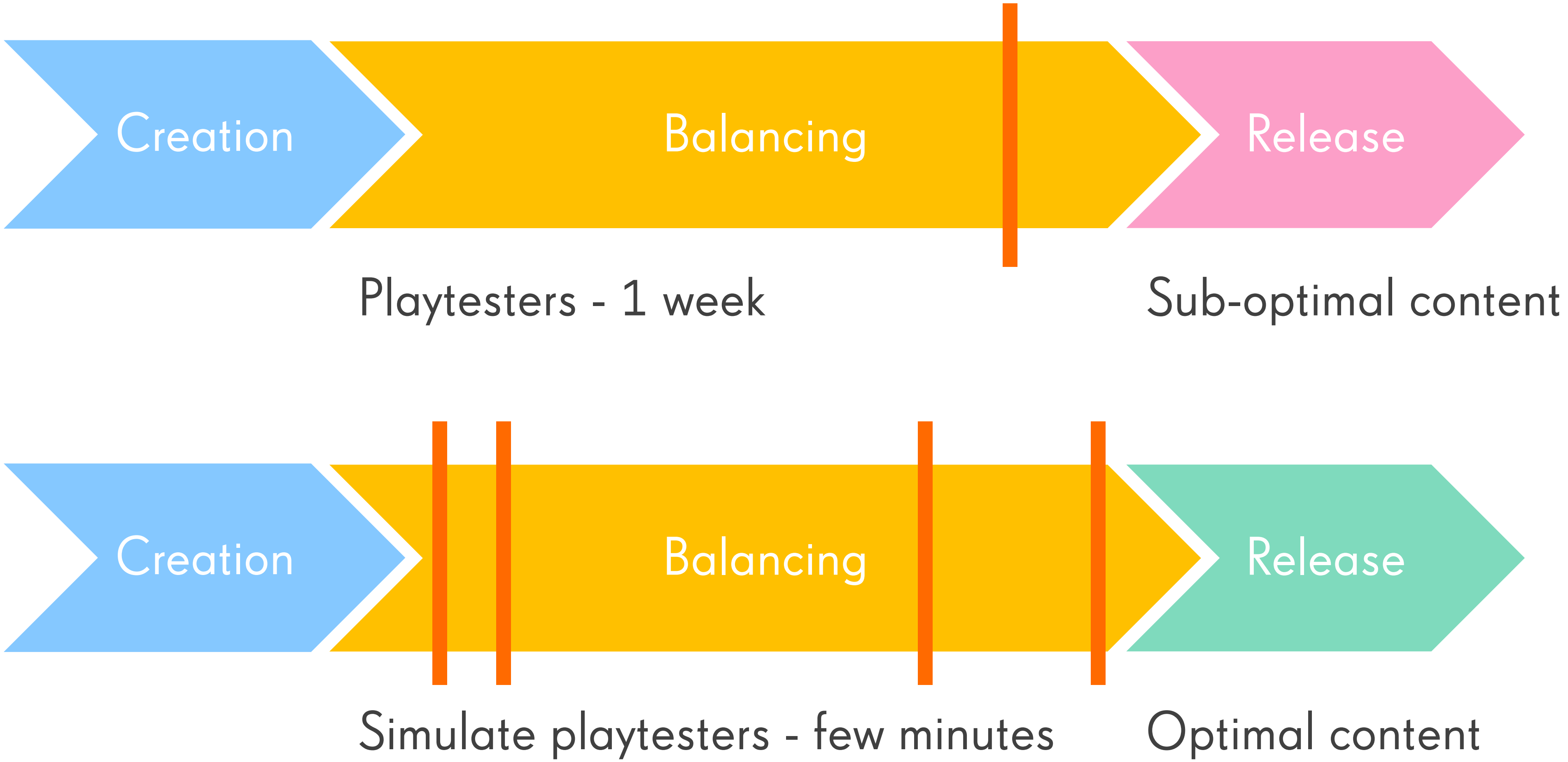
Creation → Balancing → Release

Creation	Balancing	Release
Create a new level	Modify and tweak	Staged release
Maximum creativity	Make enjoyable for everyone	Accessible to players

Tweak

Get feedback

Content production pipeline



Business benefits

Faster production pipeline

- Playtest in a few minutes
- Less context switching

Better content quality

- Balance before release
- More iterations

Harder to break the game

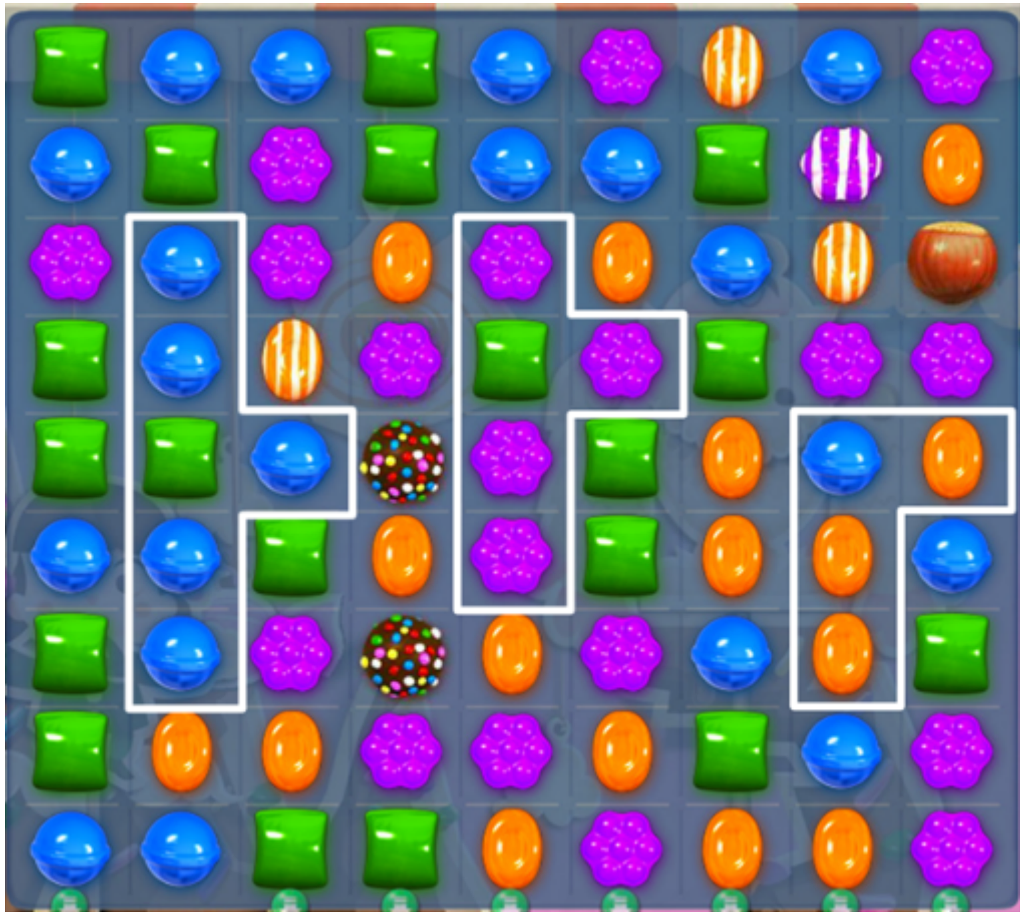
- Internal testing
- Regression testing

Stronger knowledge

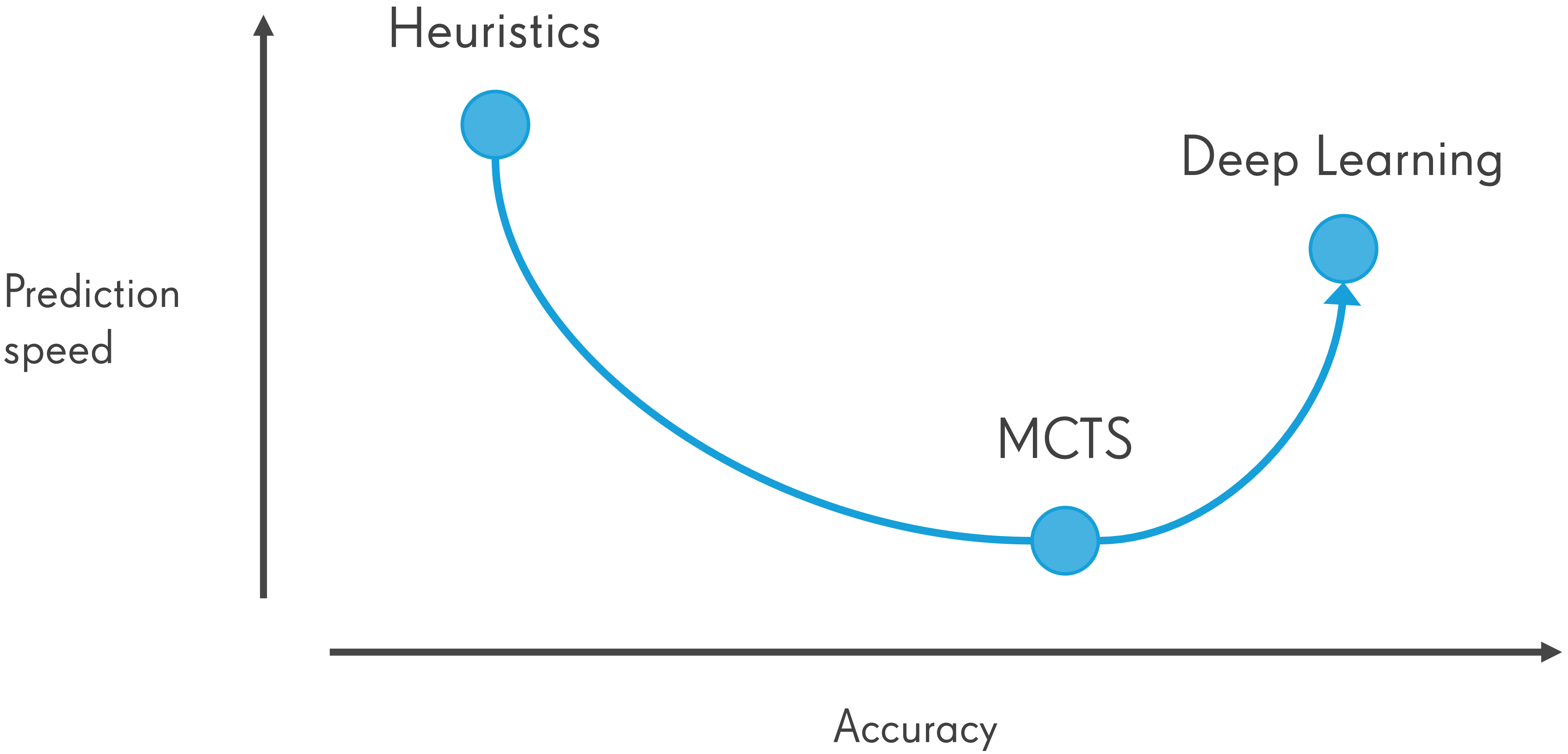
- Objective metrics
- More measures available

Player simulation

Simulating gameplay



Which approach?



Deep Learning for image classification

Dog 6%

Cat 91% Top

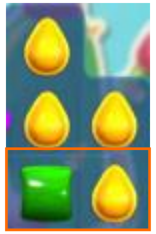
Moose 2%

Whale 1%

Source: Wikimedia Commons

Deep Learning on Candy Crush

State
Observed by human

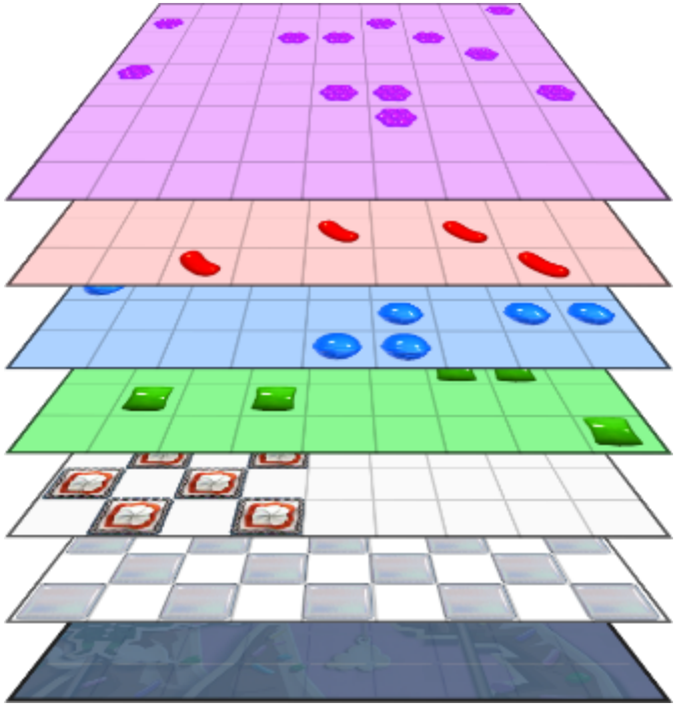


Action
Made by human

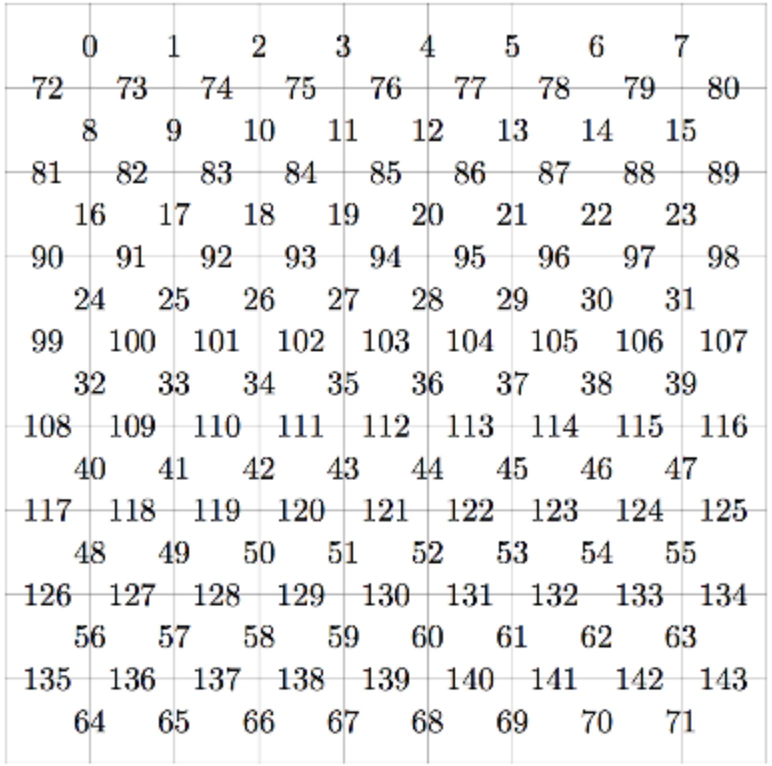
Supervised learning
Cloud Machine Learning

State encoding

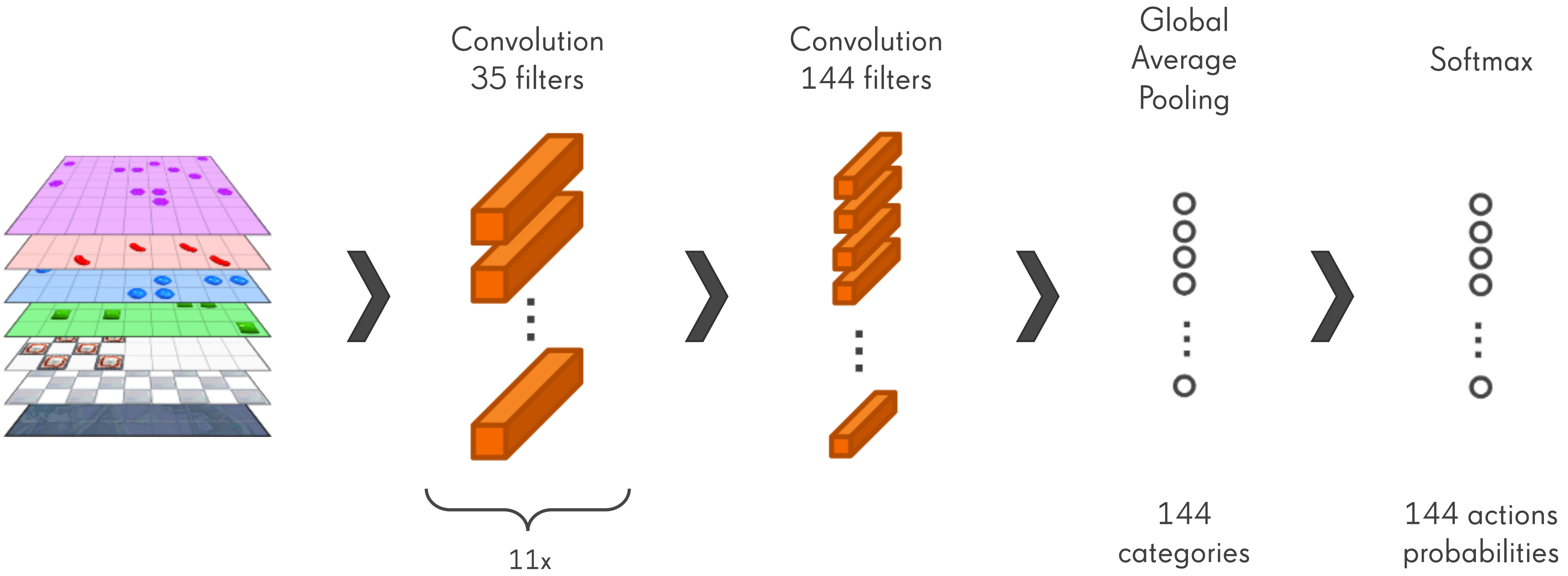
100+ binary feature layers



Action encoding



Deep network architecture

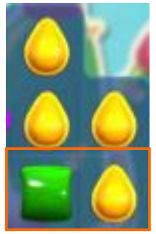


Deep learning on Candy Crush

New state

Supervised learning
Cloud Machine Learning
Trained

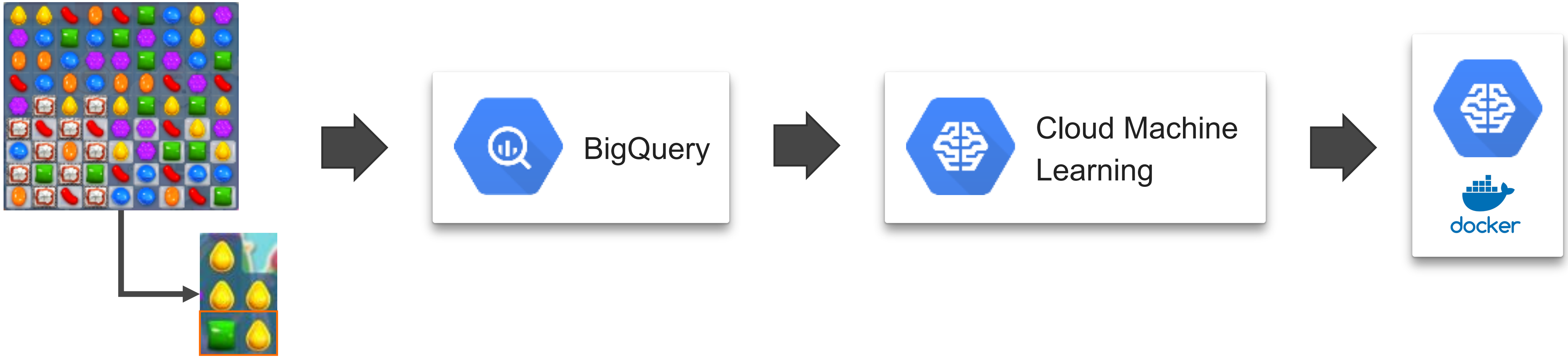
5%



94%
Most human-like move

1%

Training pipeline



Track
State-action pairs

Ingest

Train

Trained model
HTTP server
Dockerized

Correlation with real players

ML Agent

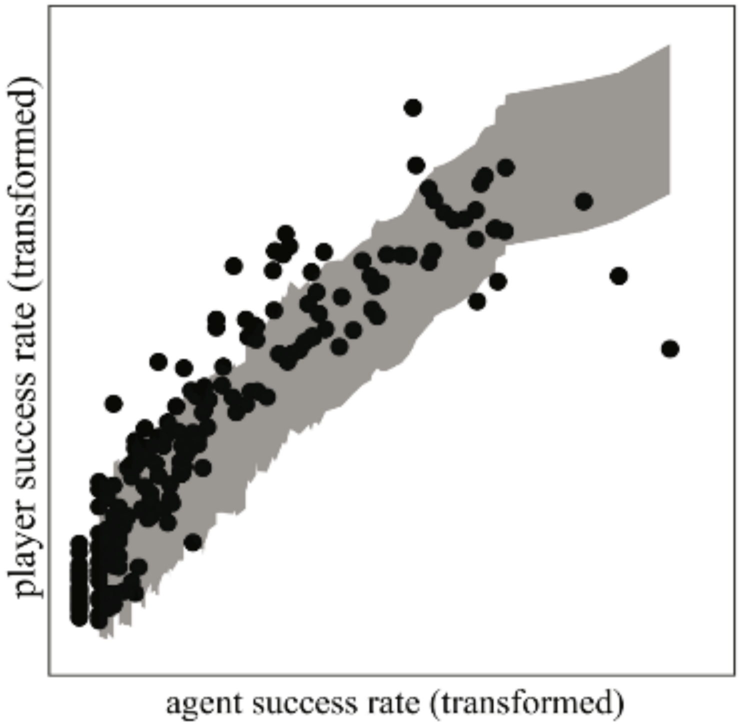
Play all levels

- ML Agent success rate

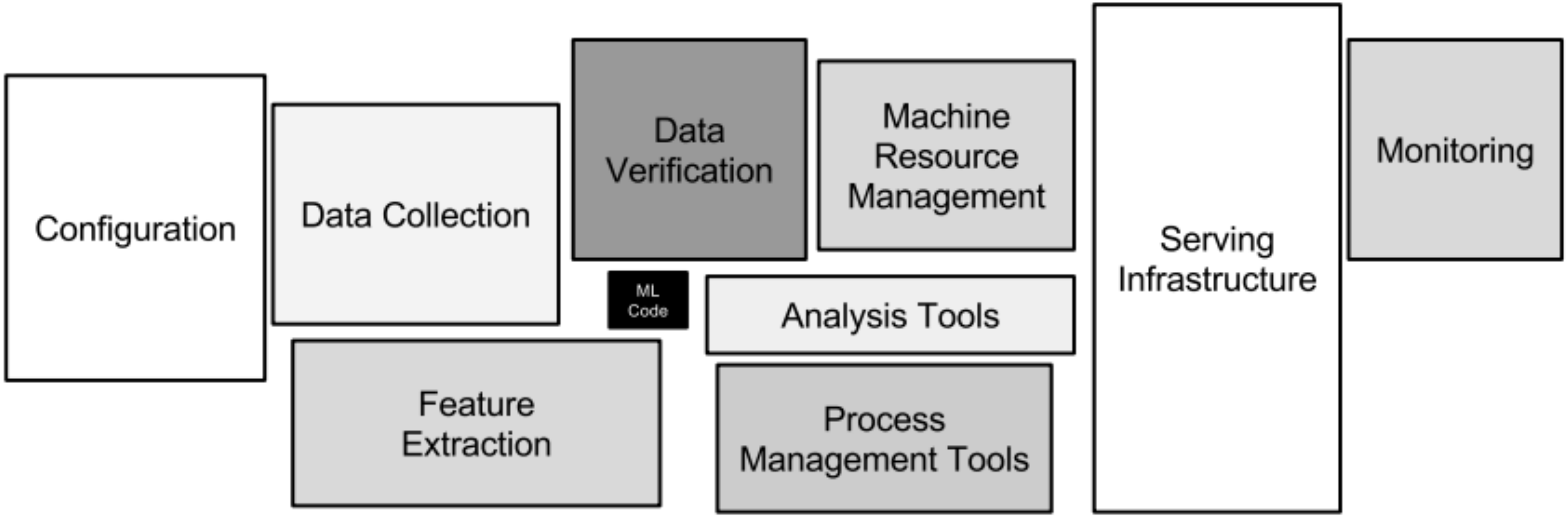
BigQuery

Historical data

- Player success rate

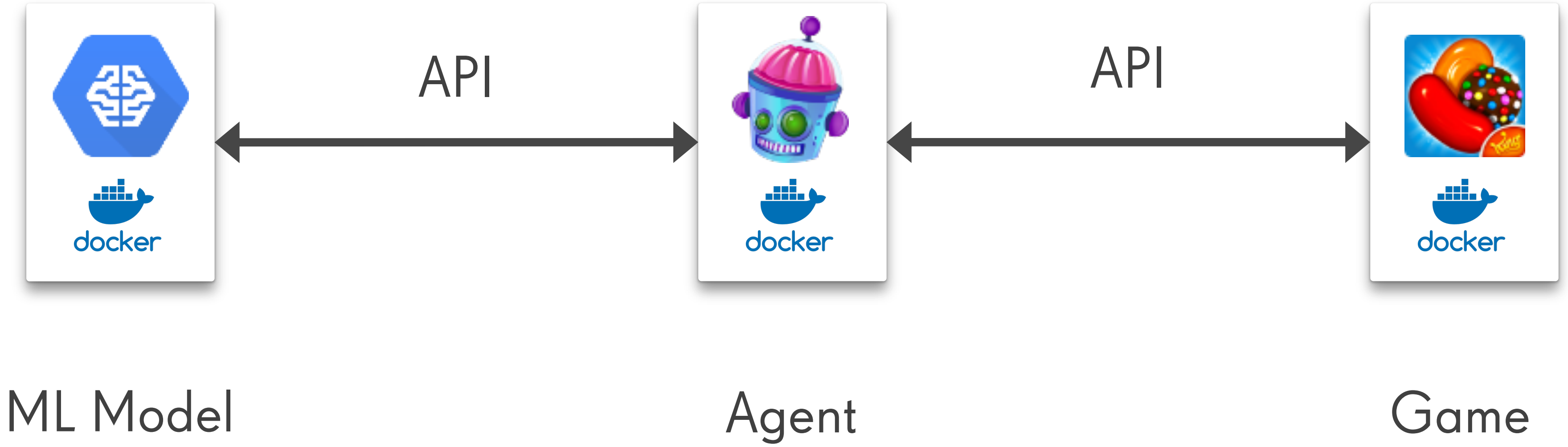


We have a model... Are we done?

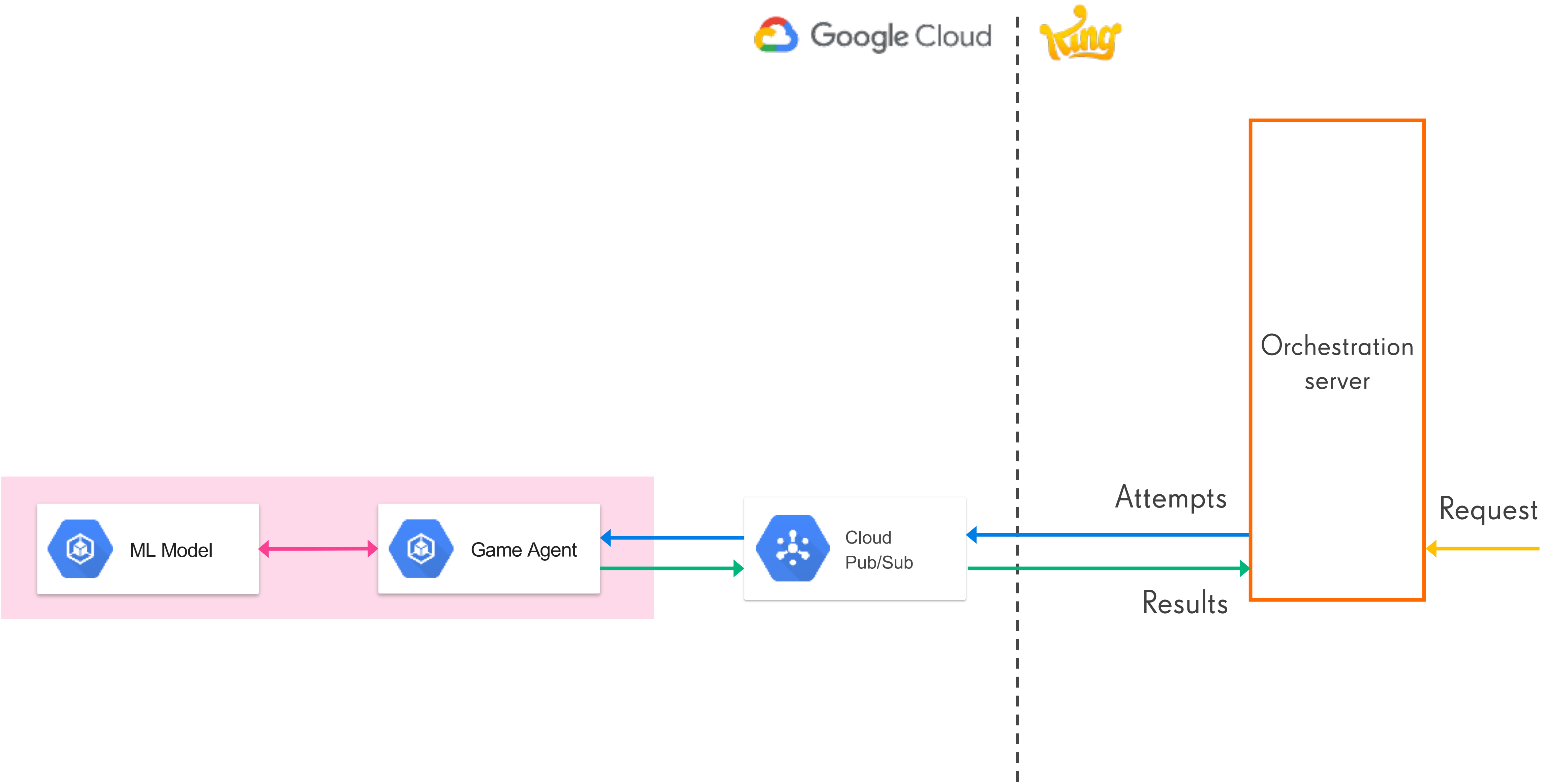


Source: Hidden Technical Debt in Machine Learning Systems, NIPS 2015, Google, Inc

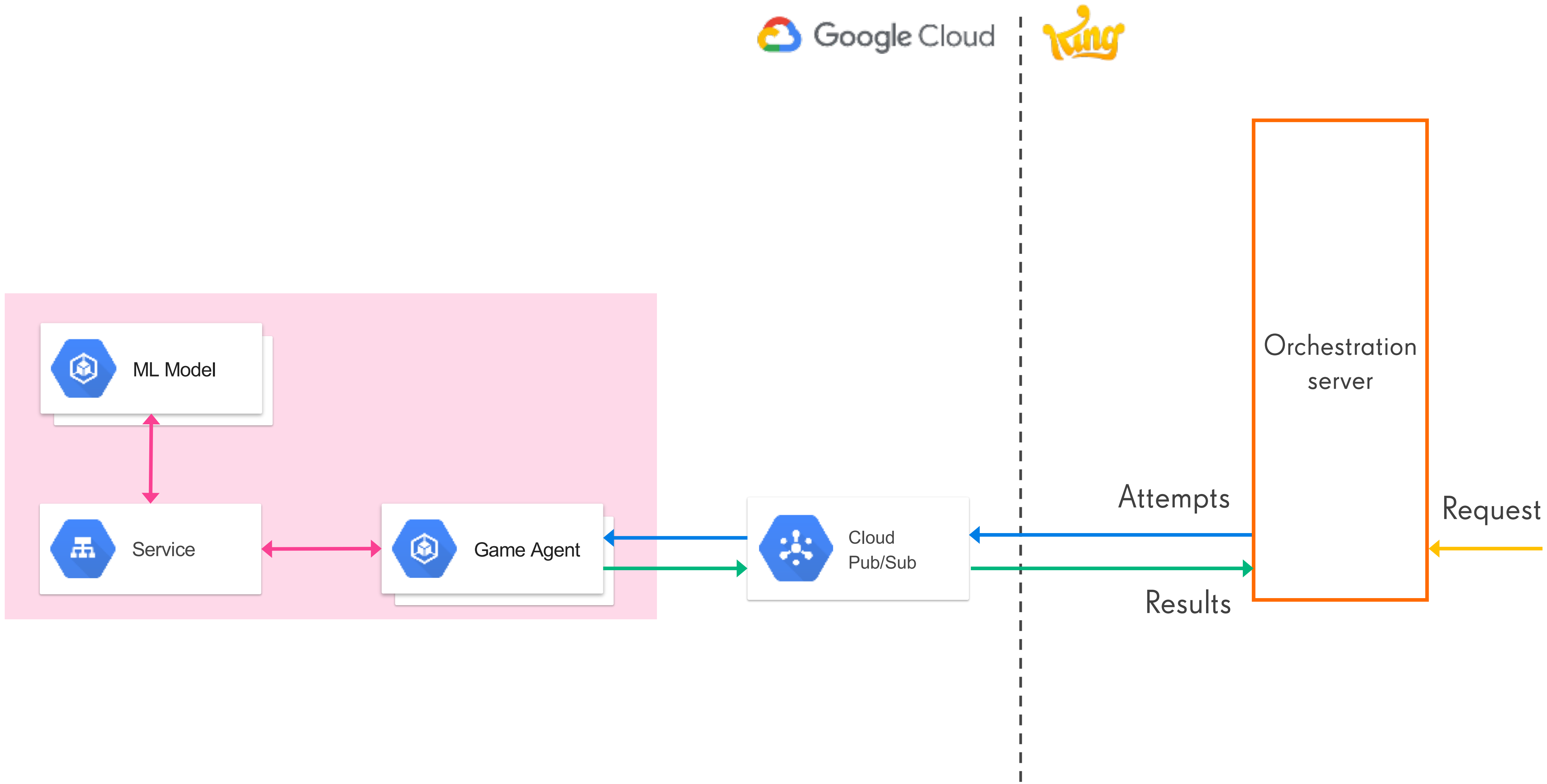
Simple setup



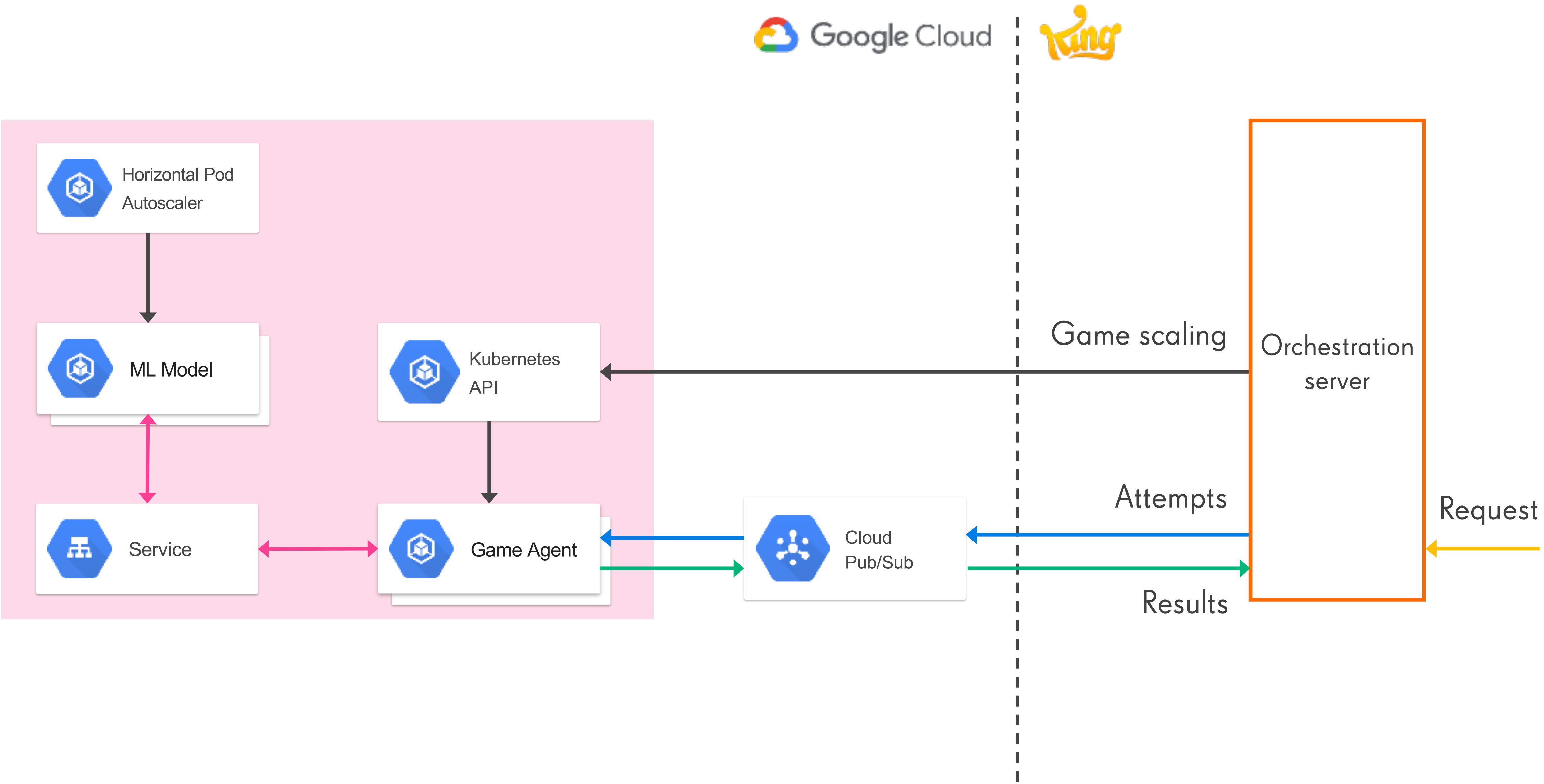
In the Cloud


Google Cloud
ML Model
Game Agent
Cloud
Pub/Sub
Attempts
Results
Orchestration
server
Request

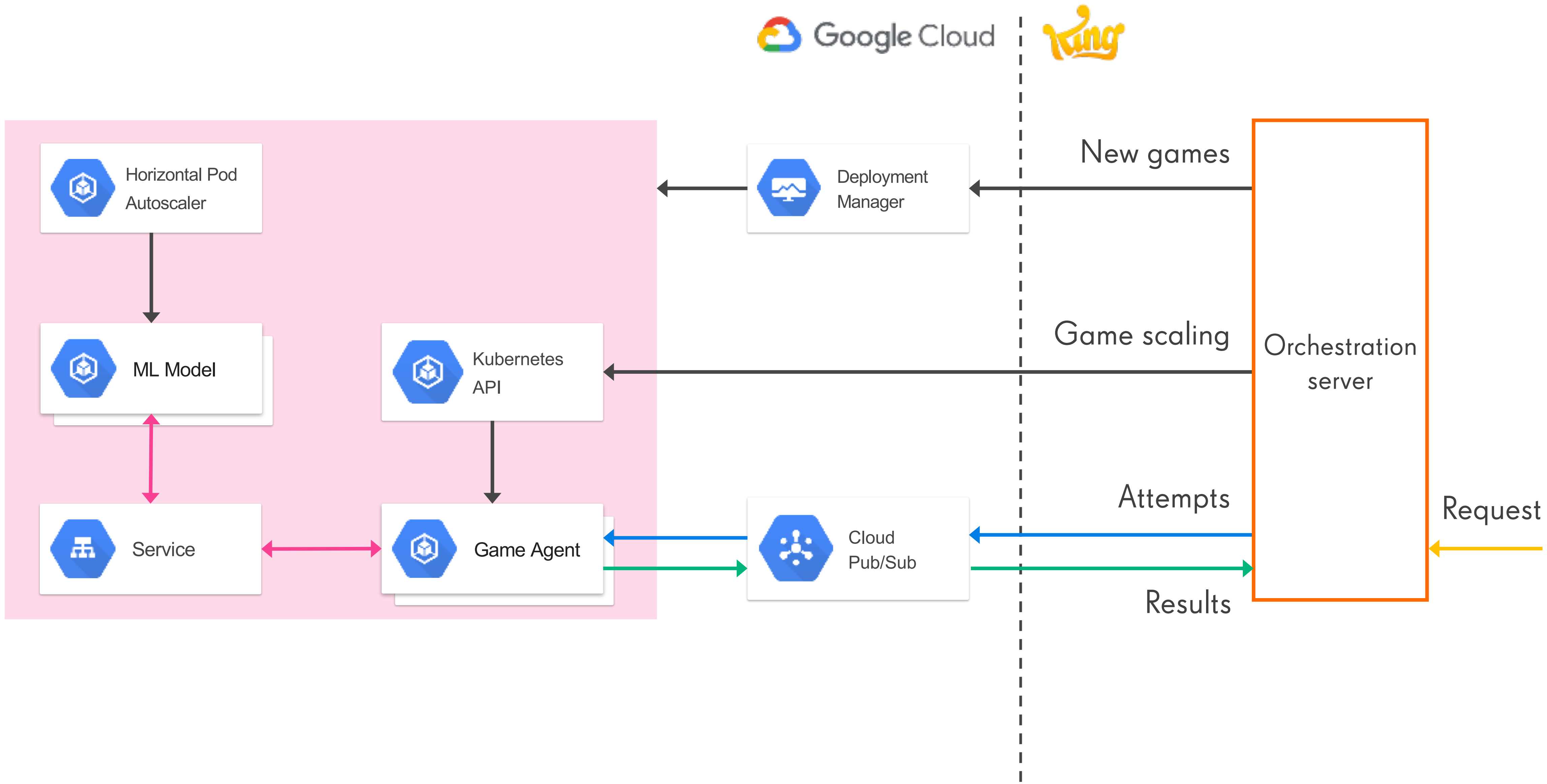
In the Cloud


Google Cloud
ML Model
Service
Game Agent
Cloud
Pub/Sub
Attempts
Results
Orchestration
server
Request

In the Cloud

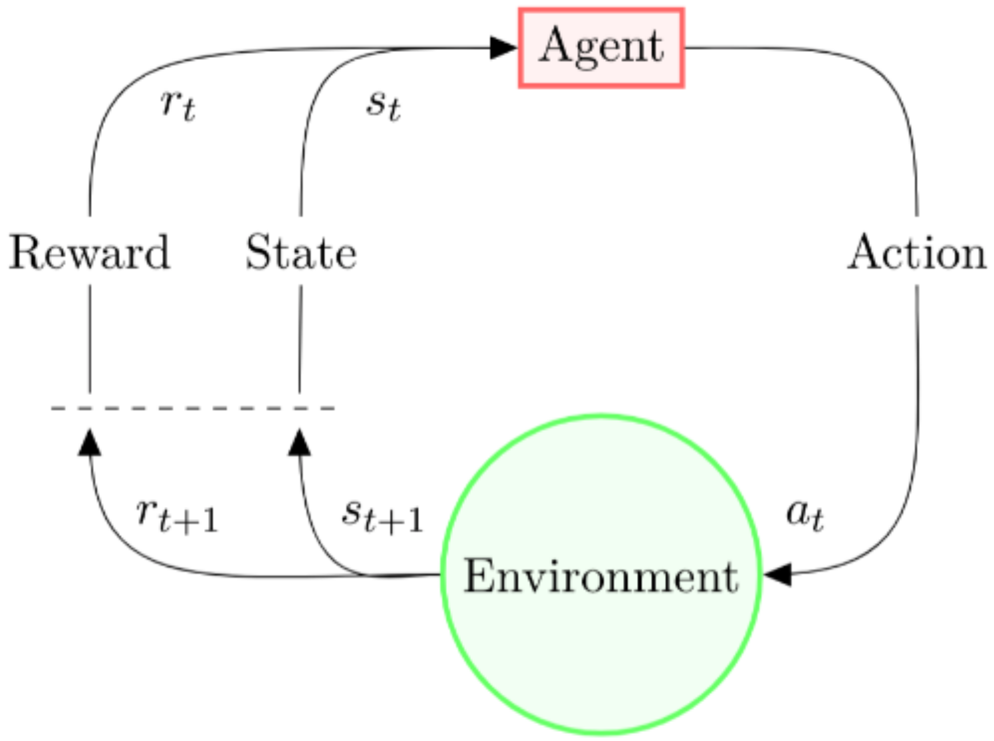

Google Cloud
Horizontal Pod Autoscaler
ML Model
Kubernetes API
Service
Game Agent
Cloud Pub/Sub
Game scaling
Attempts
Results
Orchestration server
Request

In the Cloud


Google Cloud
King
Horizontal Pod Autoscaler
Deployment Manager
New games
ML Model
Kubernetes API
Game scaling
Orchestration server
Attempts
Request
Service
Game Agent
Cloud Pub/Sub
Results

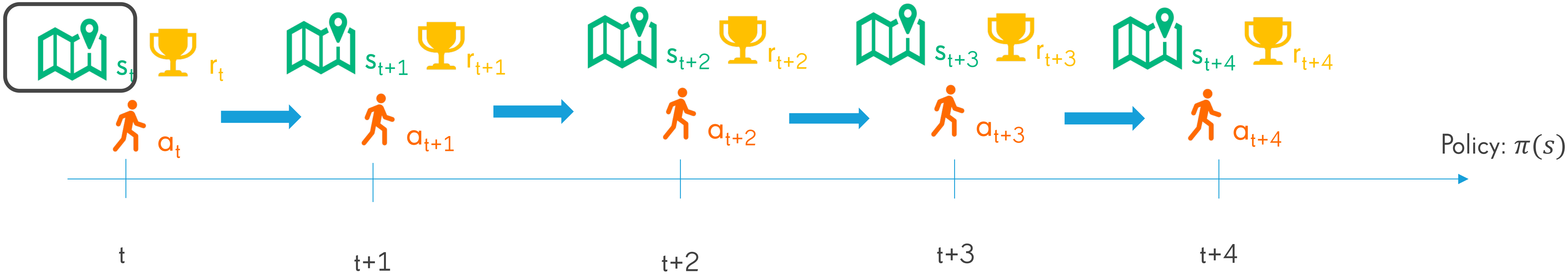
What if there is no player data available?

Reinforcement learning



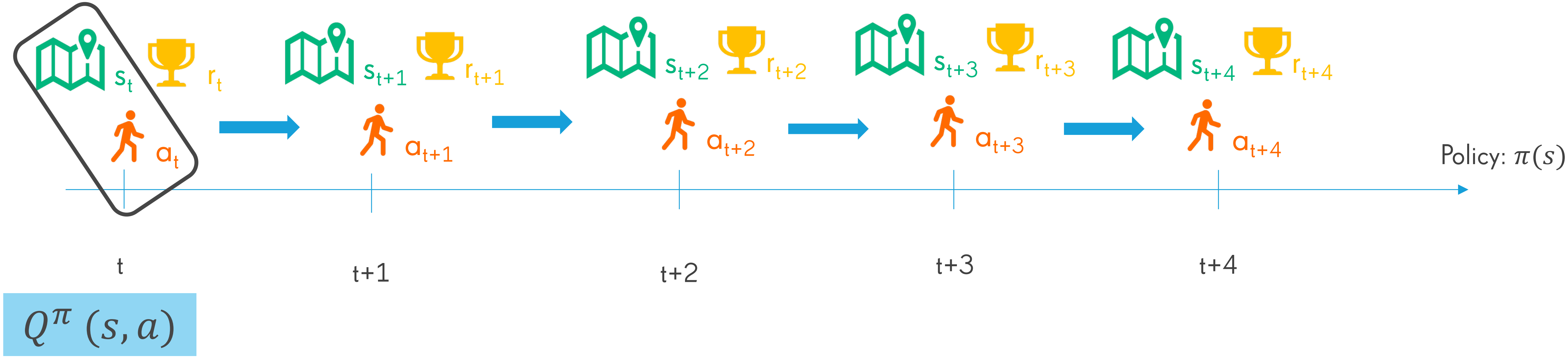
Reinforcement learning

$V^{\pi}(s)$



A. Karnsund (2019). Deep Q-Learning Tackling the Game of Candy Crush Friends - A Reinforcement Learning Approach.

Reinforcement learning



A. Karnsund (2019). Deep Q-Learning Tackling the Game of Candy Crush Friends - A Reinforcement Learning Approach.

Deep Q-Network (DQN) + extensions

Experience: (s, a, r, s')

Replay Buffer

Collect mini-batch

Batch of experiences

Behaviour-Policy π

Select action a

ε-greedy
action selection

$$Q^{\pi}(s,a) = \mathbb{E}[R_t | s_t = s, a_t = a, \pi]$$

Optimize Q-Value

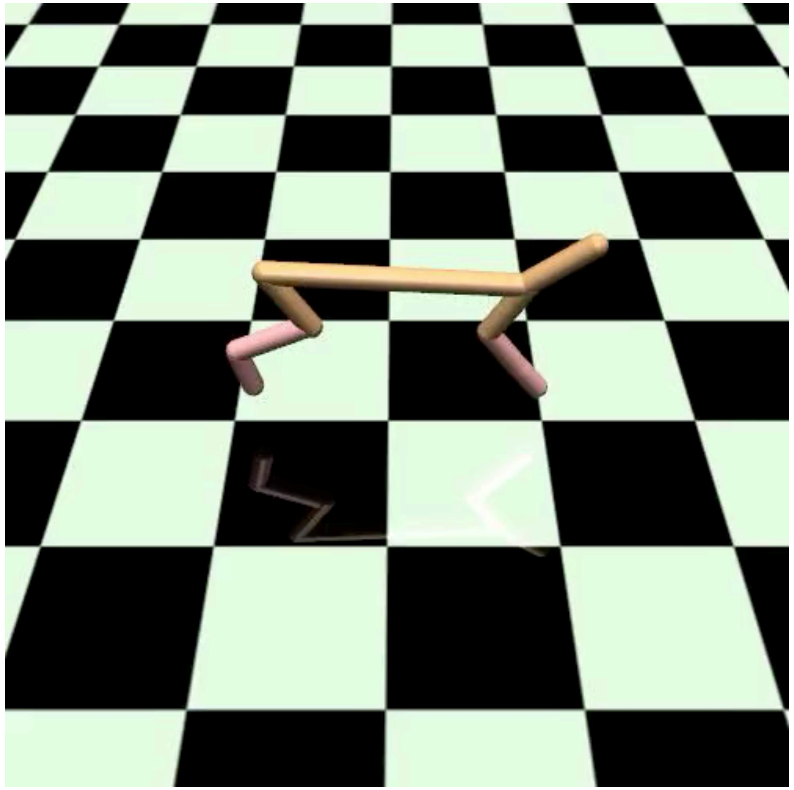
A. Karnsund (2019). Deep Q-Learning Tackling the Game of Candy Crush Friends - A Reinforcement Learning Approach.

Challenges

- Reward selection
- Generalization
- Computational complexity
- Application

Setting rewards right can be tricky

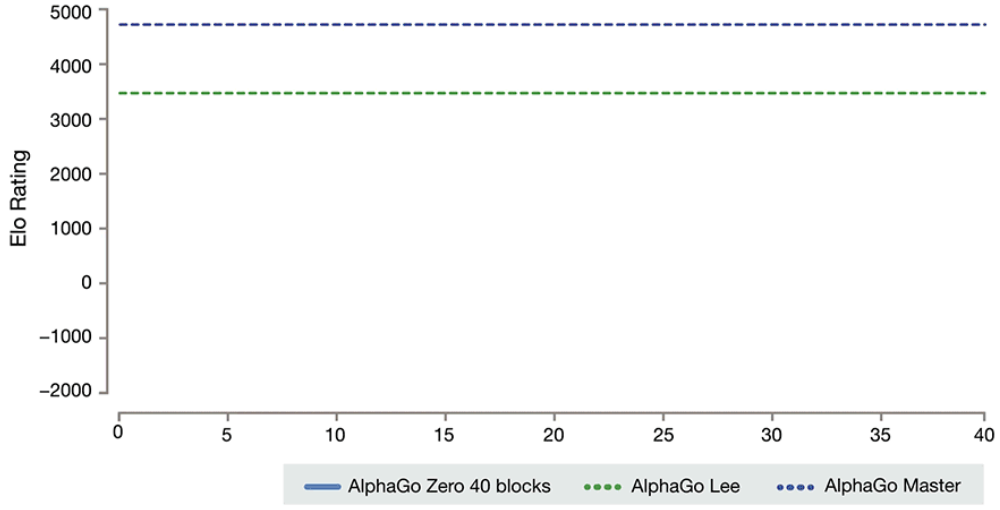
Choice of reward should reflect game's goal

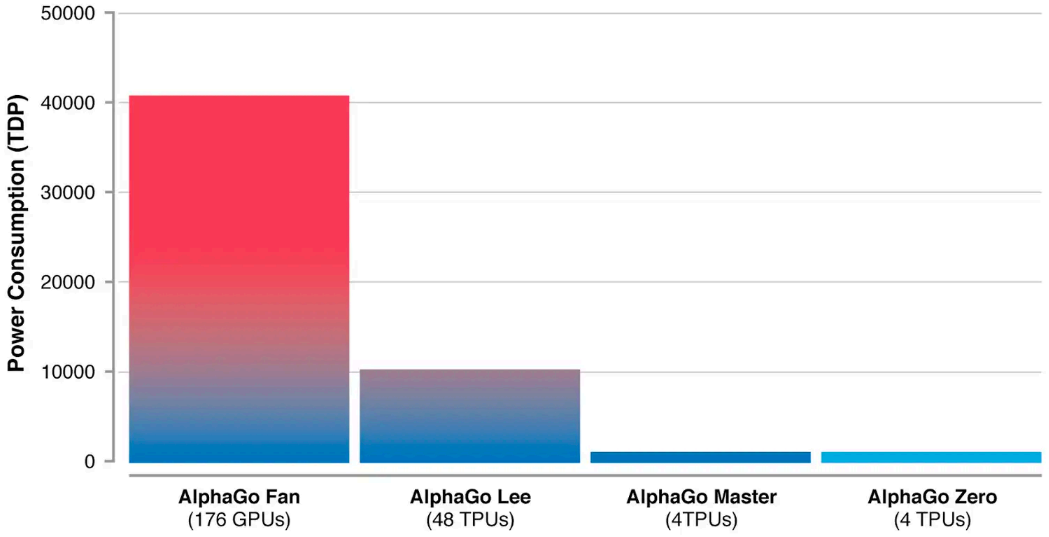


Avoid local optima

Left: https://openai.com/blog/faulty-reward-functions/
Right: https://gym.openai.com/envs/#mujoco

Computational Complexity

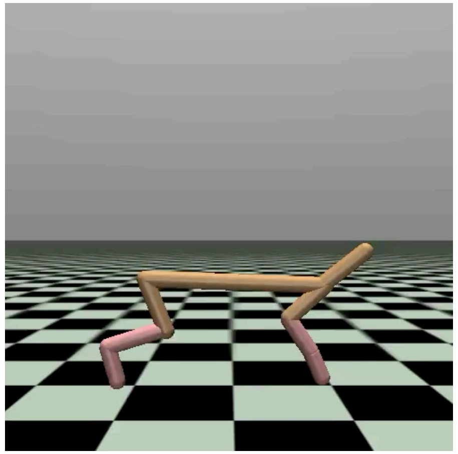




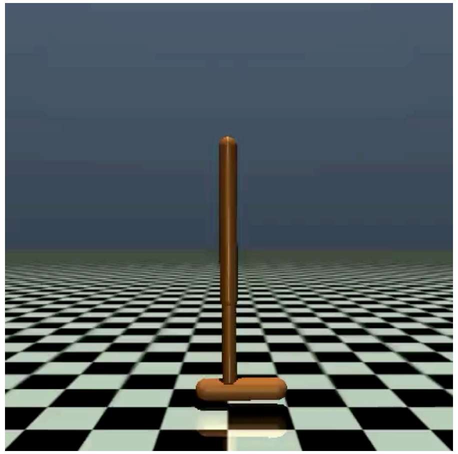
Alphago has become progressively more efficient thanks to hardware gains and more recently algorithmic advances.

https://deepmind.com/blog/article/alphago-zero-starting-scratch

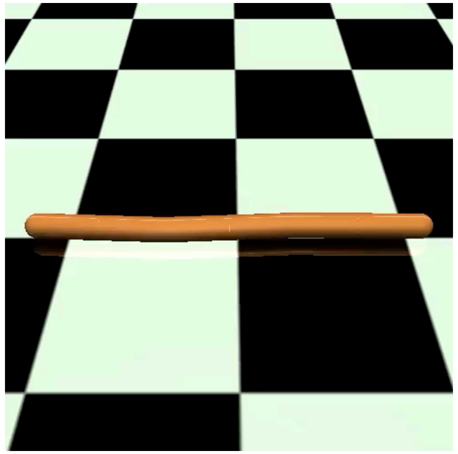
Choice of policy method and application



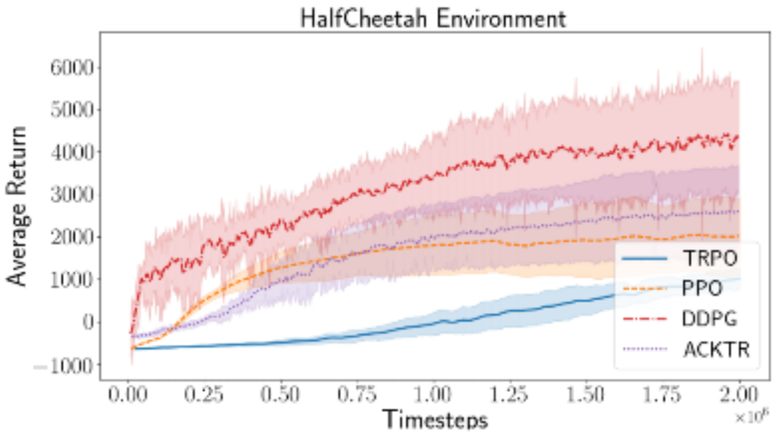
HalfCheetah

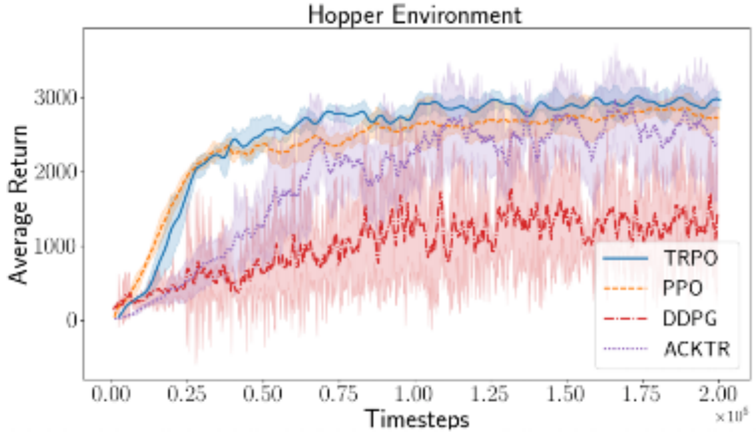


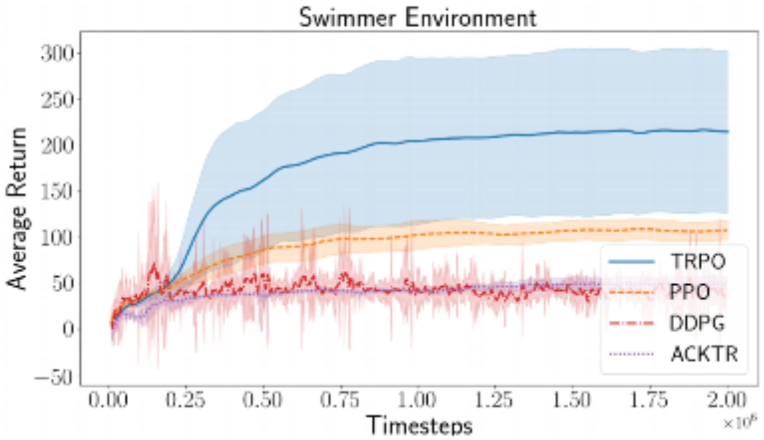
Hopper



Swimmer

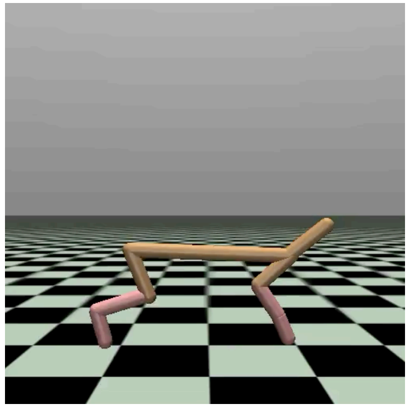


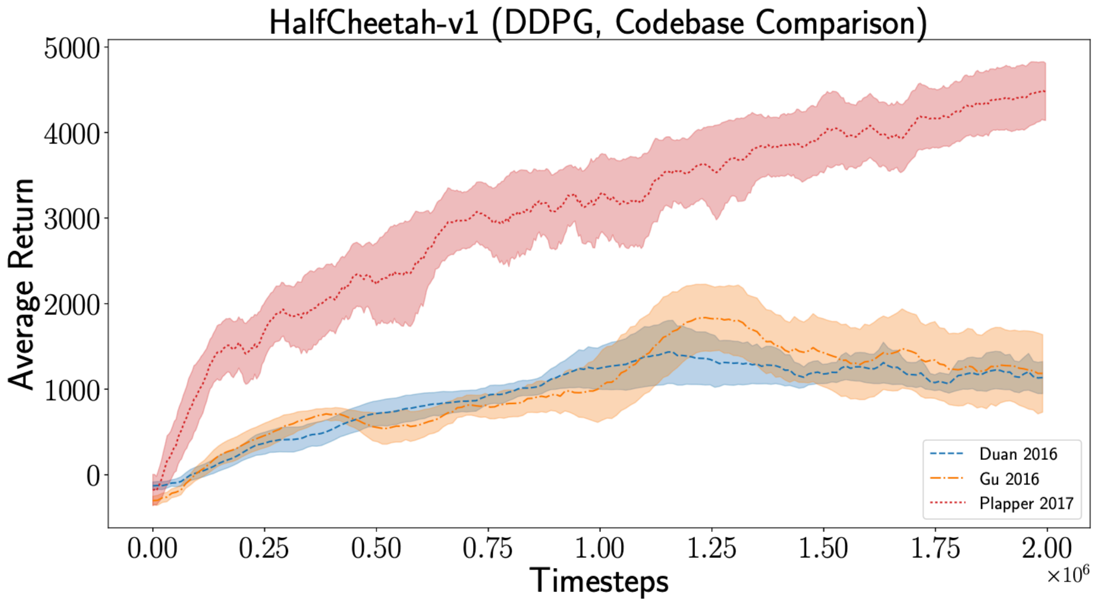


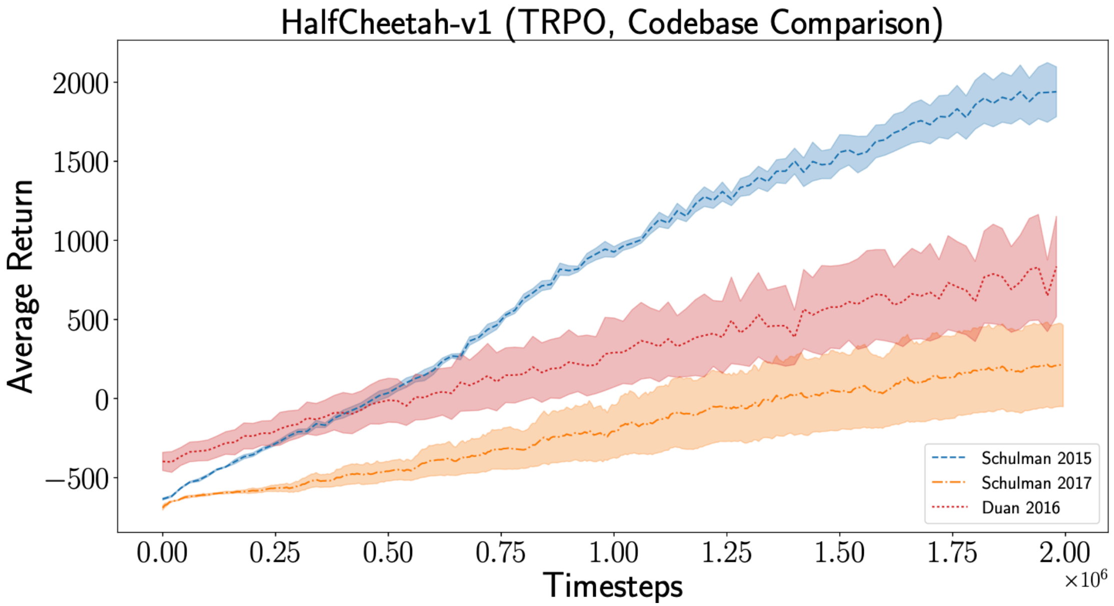


Top: https://gym.openai.com/envs/#mujoco
Bottom: Deep reinforcement learning that matters. https://arxiv.org/pdf/1709.06560.pdf

Implementation and reproducibility







Half Cheetah: Top: https://gym.openai.com/envs/#mujoco
Deep reinforcement learning that matters. https://arxiv.org/pdf/1709.06560.pdf

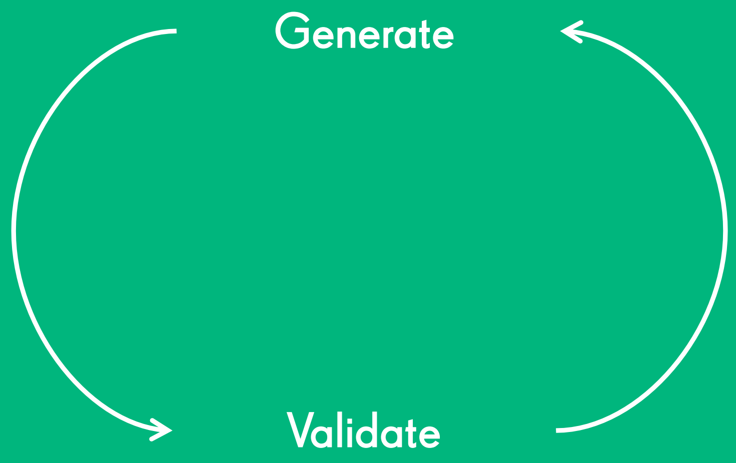
What's next?

Validation

- Improvements in the reinforcement learning based bot
- AI friendly game interfaces

Generation

- Explore content generation methods
- Complete the content generation and validation loop.
- **Assistive tool** for content generation



Want to go deeper?

https://medium.com/@TechKing

- **S. F. Gudmundsson, et al., "Human-Like Playtesting with Deep Learning",**
- A. Karnsund (2019). Deep Q-Learning Tackling the Game of Candy Crush Friends - A Reinforcement Learning Approach.
- M. Fischer (2019). Using Reinforcement Learning for Games with Nondeterministic State Transitions.
- D. Anghileri (2018). Using Player Modeling to Improve Automatic Playtesting.
- R. Ahn (2018). Cluster Analysis from a Game Theoretical Framework.
- M. Adamsson (2018). Curriculum Learning for Increasing the Performance of a Reinforcement Learning Agent in a Static First-Person Shooter Game.
- P. Eisen (2017). Simulating Human Game Play for Level Difficulty Estimation with Convolutional Neural Networks.
- S. Purmonen (2017). Predicting Game Level Difficulty Using Deep Neural Networks.
- E. R. Poromaa (2017). Crushing Candy Crush : Predicting Human Success Rate in a Mobile Game using Monte-Carlo Tree Search.
- A. Nodet (2016). Automated Heuristics in Candy Crush Saga using NeuroEvolution of Augmenting Topologies.

How to get in touch?

Questions and Collaborations

ai-rnd@king.com

Thank you! :)