2019 ICML note

Ju Yang

06/10/2019 - 06/15/2019

Long Beach, CA

Table of Contents

[Tutorial: Never-Ending Learning 1](#_Toc11677981)

[Tutorial: Meta-Learning: from Few-Shot Learning to Rapid Reinforcement Learning 2](#_Toc11677982)

[Tutorial: Algorithm configuration: learning in the space of algorithm designs 3](#_Toc11677983)

[Workshop: Automated Machine Learning 4](#_Toc11677984)

[Grey-box Bayesian Optimization for AutoML 4](#_Toc11677985)

[**Peter Frazier @Uber** 4](#_Toc11677986)

[Lessons Learned from Helping 200,000 non-ML experts use ML 5](#_Toc11677987)

[**Rachel Thomas @fast.ai** 5](#_Toc11677988)

[A Boosting Tree Based AutoML System for Lifelong Machine Learning 5](#_Toc11677989)

[An Overview of Google's Work on AutoML and Future Directions 5](#_Toc11677990)

[**Jeff Dean @ Google** 5](#_Toc11677991)

[Towards Semi-Automated Machine Learning 6](#_Toc11677992)

[**Charles Sutton @ Google research** 6](#_Toc11677993)

[Panel discussion 6](#_Toc11677994)

[Workshop: Machine Learning for Music Discovery 6](#_Toc11677995)

[From Listening to Watching, A Recommender Systems Perspective 7](#_Toc11677996)

[Characterizing Musical Correlates of Large-Scale Discovery Behavior @ Stanford 8](#_Toc11677997)

[NPR: Neural Personalised Ranking for Song Selection @Apple 9](#_Toc11677998)

[Personalization at Amazon Music 10](#_Toc11677999)

[A Model-Driven Exploration of Accent Within the Amateur Singing Voice 10](#_Toc11678000)

[What’s Broken in Music Informatics Research? Three Uncomfortable Statements @Adobe 11](#_Toc11678001)

[Workshop: Learning and Reasoning with Graph-Structured Representations 11](#_Toc11678002)

[RL with graph: chemistry and beyond 11](#_Toc11678003)

[Geometric processing 12](#_Toc11678004)

[Workshop: Self-supervised learning workshop 12](#_Toc11678005)

[Supervised learning 13](#_Toc11678006)

[Data Shapley: Equitable Valuation of Data for Machine Learning 13](#_Toc11678007)

[Feature Grouping as a Stochastic Regularizer for High-Dimensional Structured Data 13](#_Toc11678008)

[Metrics optimized example weights 13](#_Toc11678009)

[Improving Model Selection by Employing the Test Data 14](#_Toc11678010)

[Clustering 14](#_Toc11678011)

[Scalable fair clustering @IBM 14](#_Toc11678012)

[DBSCAN++: Towards fast and scalable density clustering 14](#_Toc11678013)

[concrete autoencoder 15](#_Toc11678014)

[node embedding 15](#_Toc11678015)

[spectral clustering to signed graph 15](#_Toc11678016)

[ordered weighted clustering 15](#_Toc11678017)

[Fair k-Center Clustering for Data Summarization 15](#_Toc11678018)

[better k-means ++ @google research 16](#_Toc11678019)

[spectral clustering for fairness constraints @georgia tech 16](#_Toc11678020)

[supervised hierarchical clustering 16](#_Toc11678021)

[Applications 16](#_Toc11678022)

[Exploiting Worker Correlation for Label Aggregation in Crowdsourcing @facebook 16](#_Toc11678023)

[Efficient Amortised Bayesian Inference for Hierarchical and Nonlinear Dynamical Systems 17](#_Toc11678024)

[A Multitask Multiple Kernel Learning Algorithm for Survival Analysis with Application to Cancer Biology 17](#_Toc11678025)

[Fast and Flexible Inference of Joint Distributions from their Marginals 17](#_Toc11678026)

[Cognitive Model Priors for Predicting Human Decisions 18](#_Toc11678027)

[Conditioning by adaptive sampling for robust design 18](#_Toc11678028)

[Direct Uncertainty Prediction for Medical Second Opinions @google brain 19](#_Toc11678029)

[Dynamic Measurement Scheduling for Event Forecasting Using Deep RL 19](#_Toc11678030)

[DeepNose: Using artificial neural networks to represent the space of odorants @ cold spring harbor 19](#_Toc11678031)

[Time series 19](#_Toc11678032)

[Generative Adversarial User Model for Reinforcement Learning Based Recommendation System 20](#_Toc11678033)

[Deep Factors with Gaussian Processes for Forecasting @Amazon AWS 20](#_Toc11678034)

[Weakly-Supervised Temporal Localization via Occurrence Count Learning 21](#_Toc11678035)

[Imputing missing events in continuous time event series 21](#_Toc11678036)

[ML System 21](#_Toc11678037)

[Fault tolerance in iterative convergent ML @Petuum 21](#_Toc11678038)

[Static automatic batching in tensorflow @ Google brain 22](#_Toc11678039)

[Reinforcement learning 22](#_Toc11678040)

[controlled dimension reduction in RL 22](#_Toc11678041)

[reparameterizable RL @salesforce research 22](#_Toc11678042)

[off-policy RL @ BOSCH 23](#_Toc11678043)

[RL in internet congestion control 23](#_Toc11678044)

[model based active exploration MAX @ NNAISENSE https://arxiv.org/abs/1810.12162 23](#_Toc11678045)

[inverse RL with ranking 23](#_Toc11678046)

[bellman GAN 23](#_Toc11678047)

[information inference 24](#_Toc11678048)

[adaptive neural tree @microsoft 24](#_Toc11678049)

[Connevtivity-optimzied representation learning 24](#_Toc11678050)

[learn to rout in similarity graph 24](#_Toc11678051)

[sparsified neighborhood mixing @google ai 25](#_Toc11678052)

[learn to grow 25](#_Toc11678053)

[Invited talk 27](#_Toc11678054)

[Learning @ cognitive science 27](#_Toc11678055)

[Best paper: challenge assumption in unsupervised learning representation 28](#_Toc11678056)

[Expo 28](#_Toc11678057)

[Cistro 28](#_Toc11678058)

[SigOpt 28](#_Toc11678059)

[Yandex 28](#_Toc11678060)

# Tutorial: Never-Ending Learning

* Organization: <http://lifelongml.org/>
* Website: <https://sites.google.com/site/neltutorialicml19/>
* Slides: <https://docs.google.com/viewer?a=v&pid=sites&srcid=ZGVmYXVsdGRvbWFpbnxuZWx0dXRvcmlhbGljbWwxOXxneDo1ZTRhMGQ3OGM5NzI0NjFm>

Content

* Reinforcement learning
* Transfer learning
* Representation learning
* Multi-task learning
* Curriculum learning
* Catastrophic Forgetting

Function learning X -> y

Agent learning: sensor => action, collections of functions

Set of functions, set of reflection

Tuple of sensor, effector, memory, Fns, Graph, L

* Learn to learn
* Unlabeled data
* Initial structure of agent

NELL: never ending learning learner

Structure for agent with unlabeled data

* NLP ontology knowledge graph
* Use deduction for unlabeled data
* Learn concept by its context

Continue to learn, First order constraints

Self-reflection

RL, SOAR：search space

Continue / sequential learning (CL）

* Sequence of tasks, T1, T2,... Tn
* Multi-task from original model (check paper)
  + Fine tuning
  + Feature
  + Joint training
* Without forgetting
* Constraint regularization
  + L2 too rigid
  + Elastic weight consolidation EWC

External knowledge (use and update)

* Memory + knowledge graph
* Memory network
  + End2end
* KG
  + Dense representation in vector space
  + KG embedding of triplet
  + KG for doc classification [NAACL 2018]
  + Use external knowledge for NLP learning

Representation learning in NLP

* States, and sequence
* Word embedding
* Word2vec, self-attention BERT ⇒ check it out

Learning to learn by gradient descent by gradient descent

<https://arxiv.org/abs/1606.04474>

curriculum -driven learning Bengio

* What is curriculum: sequence of tasks

Curiosity-driven Learning

* Predict reward of action

# Tutorial: Meta-Learning: from Few-Shot Learning to Rapid Reinforcement Learning

* Slides: <https://drive.google.com/file/d/1DuHyotdwEAEhmuHQWwRosdiVBVGm8uYx/view?usp=sharing>
* Additional reading:
  + <https://towardsdatascience.com/advances-in-few-shot-learning-a-guided-tour-36bc10a68b77>
  + <https://zhuanlan.zhihu.com/p/61215293>

Definition:

Meta-learning application in supervised learning ⇒ few shot learning

No large dataset, few samples for each class

Black-box adaption

* Output some parameters of the network

Meta learning v.s. Learn to learn

Infer all parameters (without training it) in a scalable way?

* Fine tune
* Optimization-based inference
* Meta-parameter???
  + Second order optimization problem
* Optimization v.s. Black-box adaption
* Bayesian graph task-specific parameters

meta-RL challenge

* Meta overfitting
  + Define task distribution
  + Too few tasks
* Memorization
  + Learn single classifier that doesn’t adapt
  + Generalize and learn fast about new objects
  + Initiate NN architecture, and generalize fast

Online meta-learning ICML 19’

<https://arxiv.org/abs/1902.08438>

# Tutorial: Algorithm configuration: learning in the space of algorithm designs

* Slides: <http://ml.informatik.uni-freiburg.de/~hutter/ICML19_AC.pdf>

Design Algo as ML problem

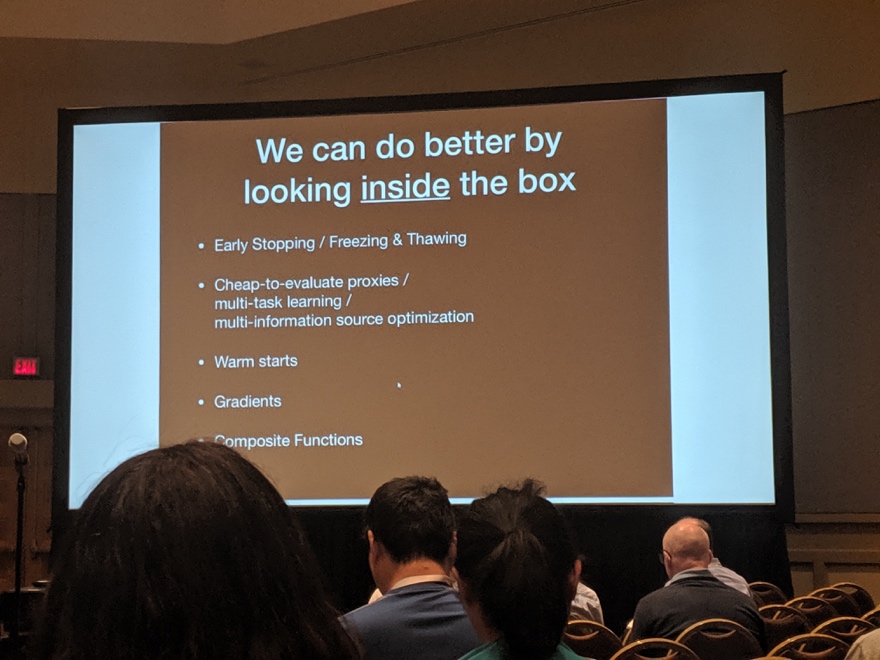
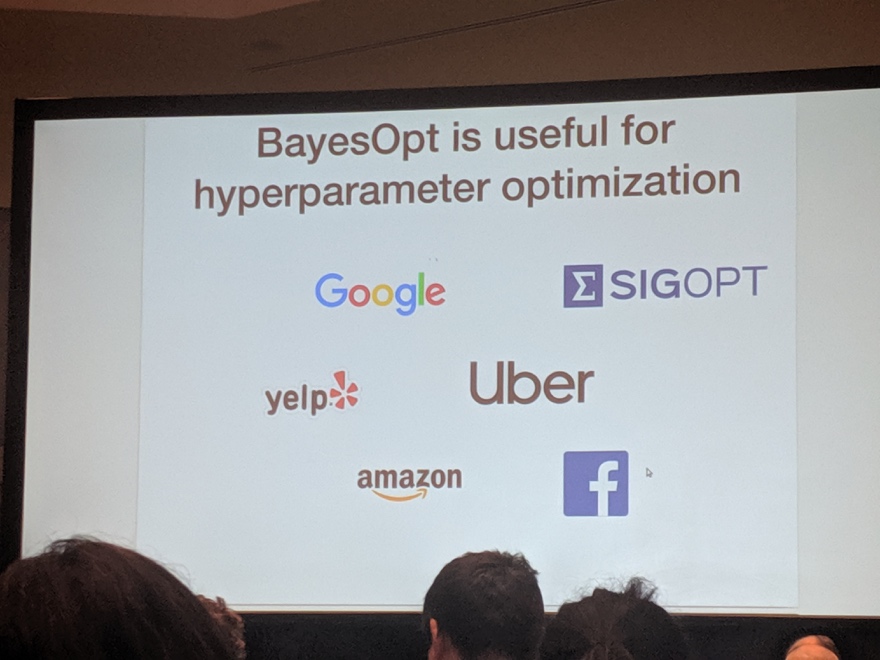
* View TSP/combinatory etc algo as ML
* Build problem solver (local search algo)
* Meta-algo: take other algos as input
* Objective
  + Run time
  + Black-box opto
  + Hyperparameter opto
* Uncertainty estimate from the regression model
* High variation run-time across algo
* SMAC (bayesian opto)

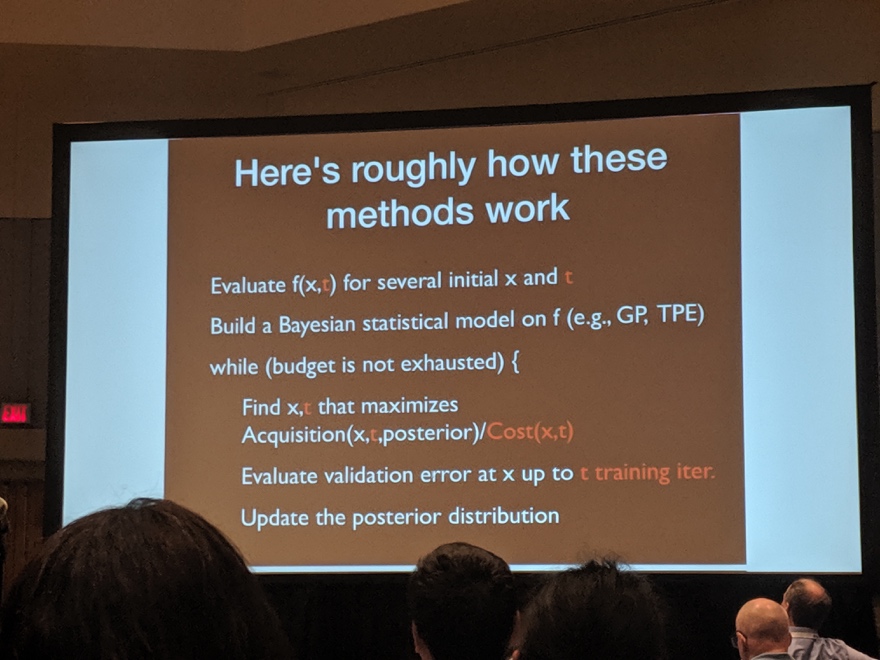
# Workshop: Automated Machine Learning

* Website: <https://sites.google.com/view/automl2019icml/>
* Book: <https://www.springer.com/gp/book/9783030053178>
* Organization: <https://www.automl.org/events/>

## Grey-box Bayesian Optimization for AutoML @Uber

Peter Frazier

[https://icml.cc/Conferences/2019/AcceptedWorkshopFAQ](https://icml.cc/Conferences/2019/AcceptedWorkshopFAQ#)



* BayesOpt in blackbox
* Classical bayesian opto: EGO
* Points evaluated
* 2012 use BO for hyperparameter tuning
  + In NN hyperparemter
  + NN architecture
  + Widely used
* Blackbox ⇒ greybox
  + Better by looking inside the box
  + Early stopping / freezing, thawing
  + Multi-task learning
  + Warm starts
  + Gradients (leverage)
* Epochs and learning curve from training
  + Model the learning curve directly
  + Such that not to perform whole evaluation
* Hyperband
  + Stop poor evaluations
* Combine BO with early stopping: active research
  + Based on training???
  + BO means Bayesian optimal stopping 2019 ICML
* Cheap eval proxies
  + Multi-task BO
  + GP bandit
* GREY BOX
  + f(x,t)
  + t: training iteration
  + Search in both acquisition of x, and t, and cost
* Inference based on learning curve
  + Tweak EI acq function
* Look into the blackbox to improve BO performance

## Lessons Learned from Helping 200,000 non-ML experts use ML @fast.ai

Rachel Thomas

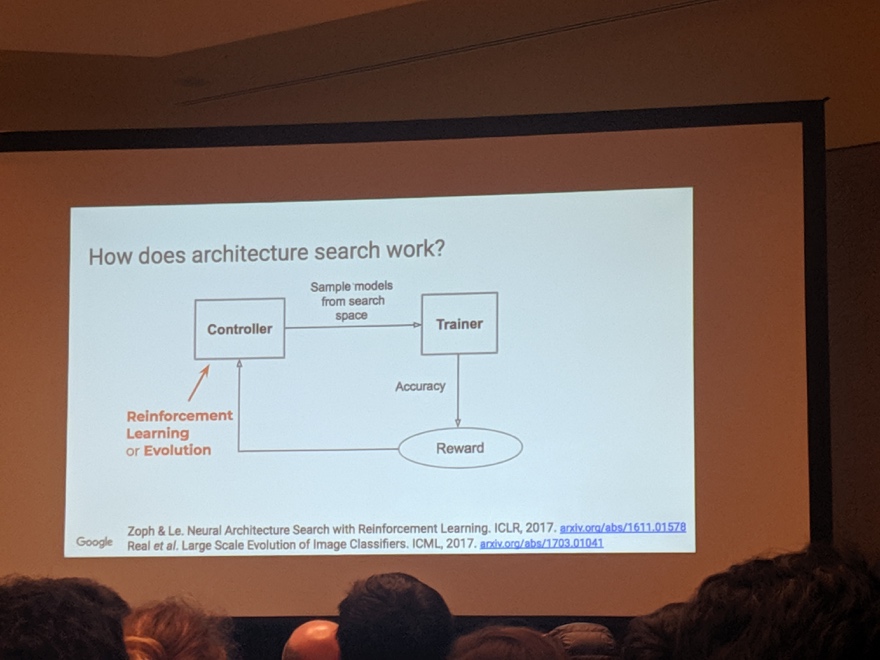
* Making nn uncool again
* Fastai libracy -> pytorch
* Use DL in application
* Productionalize / monitor the work
* ML: the high interest credit card of technical debt
  + <https://ai.google/research/pubs/pub43146>
  + <https://storage.googleapis.com/pub-tools-public-publication-data/pdf/43146.pdf>
* Learning rate finder
* Transfer learning ⇒ applied to NLP tasks / image
  + With smaller dataset
* Courses
  + <https://course.fast.ai/>
  + Test applications for my own side projects
* Start with problem and people, and then search for solutions

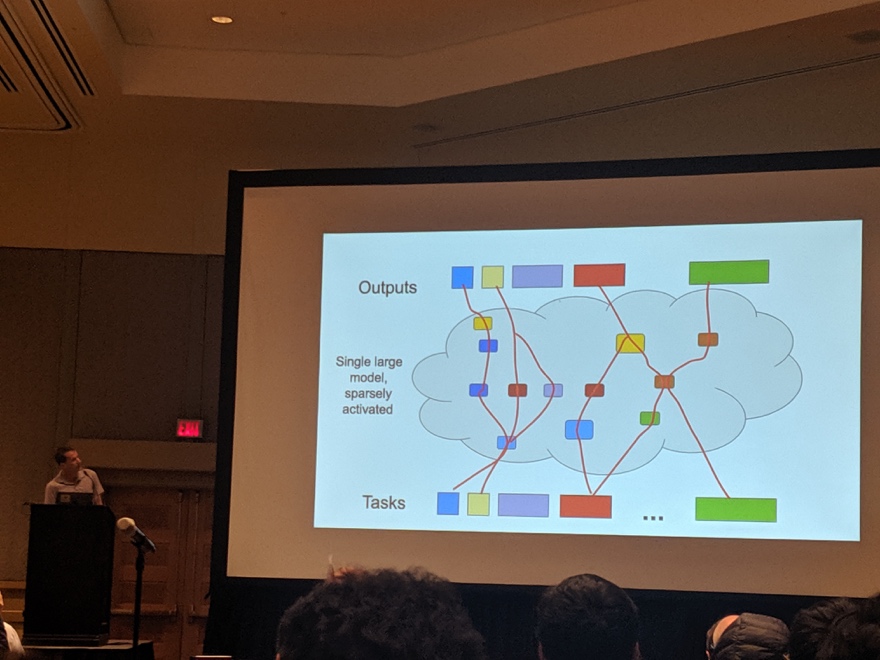
## A Boosting Tree Based AutoML System for Lifelong Machine Learning

* Neurips 2018 automl challenge
* High cardinality features
  + Streaming encoder
  + TF: Frequency encoding (probability)
  + Ordinal representation
* High order feature combo
* Feature selection
  + Expansion-reduction
* Concept drift
* Hyper-Opto
  + Portfolio search + BO in local search space
  + Search with warm start

## An Overview of Google's Work on AutoML and Future Directions @Google

Jeff Dean





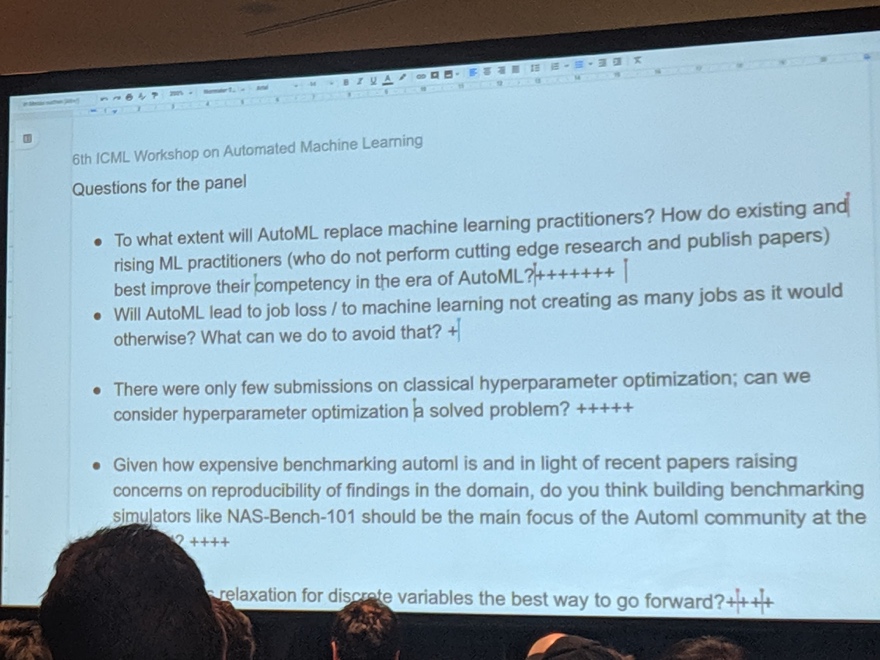
* 20 yrs in google, mapreduce, tensorflow
* Design choices ⇒ fully automated
* autoML on tabular data at Google
* Architecture search
* Activation function search
* autoRL
* Current autoML
  + Data + compute
  + Cons: cold start, no prior knowledge
  + Solutions: transfer learning / meta-learning / multi-task learning
* Future
  + Bigger model, sparsely activated
    - Per-example routing
    - MoE layer
  + Single model to solve multiple tasks
    - Generalize
  + Dynamically learn / grow pathways
    - Pathway search
  + Model architecture adapt
  + New tasks use existing skills
* Problems
  + Catastrophic forgetting
  + Balance obj across tasks
  + Dynamic routing of experts
  + large-scale/dynamic modeling ← software / hardware
  + Meta-level capacity control

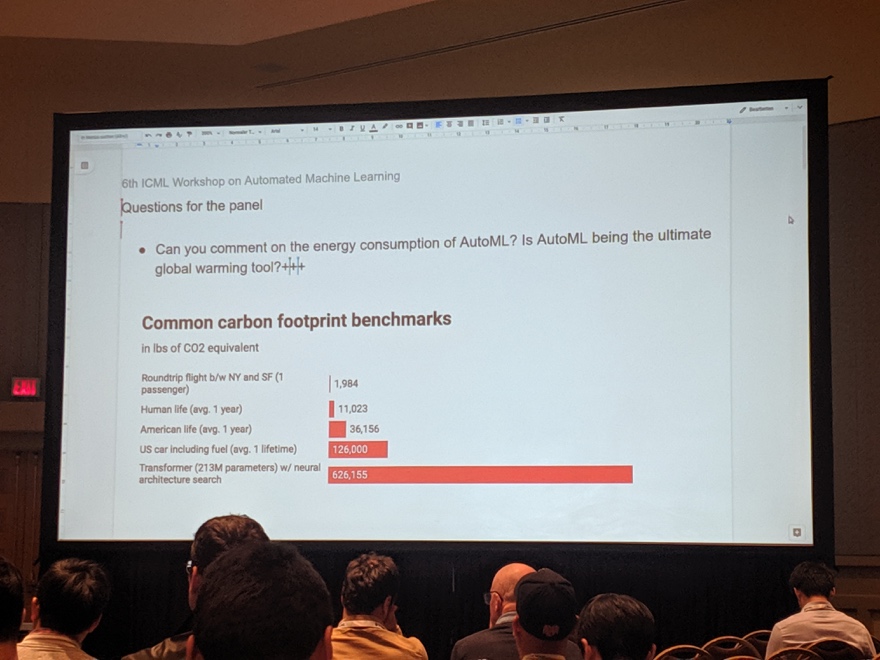
## Towards Semi-Automated Machine Learning @ Google research

Charles Sutton

* Automl is more than just models
  + At the system level, only a very small piece is ML
* Semi-AutoML
  + AI assistant
  + Design issues
* Data base/data mining/programming
* AIDA
  + AI for data analytics
  + Transformation / integration
    - Parse csv

## Panel discussion





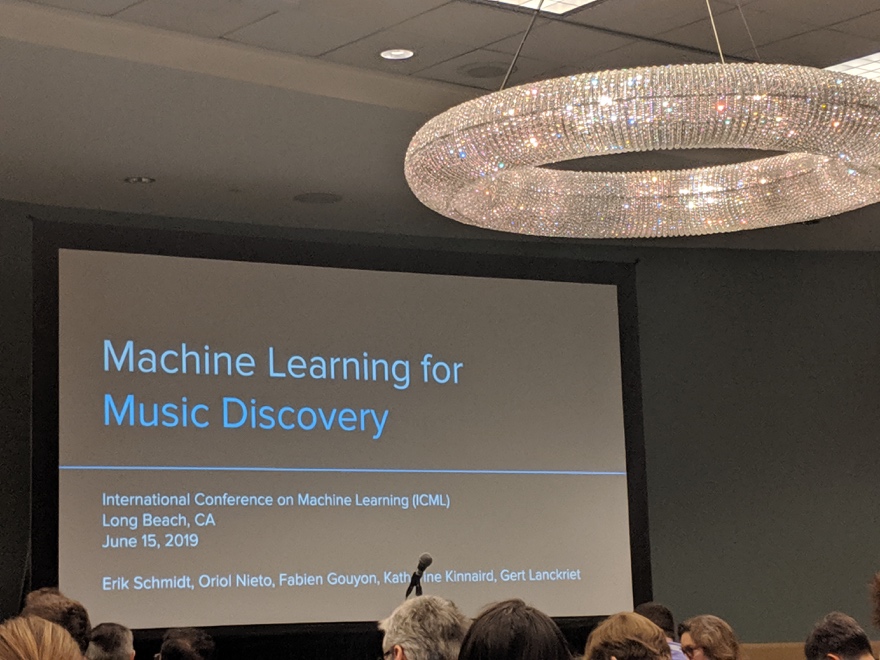
* AutoML, machine learning practitioner, and job at risk
* Deep AutoML and energy consumption
* Classical hyperperamter search (grid, random, etc) v.s. Bayesian-based approach

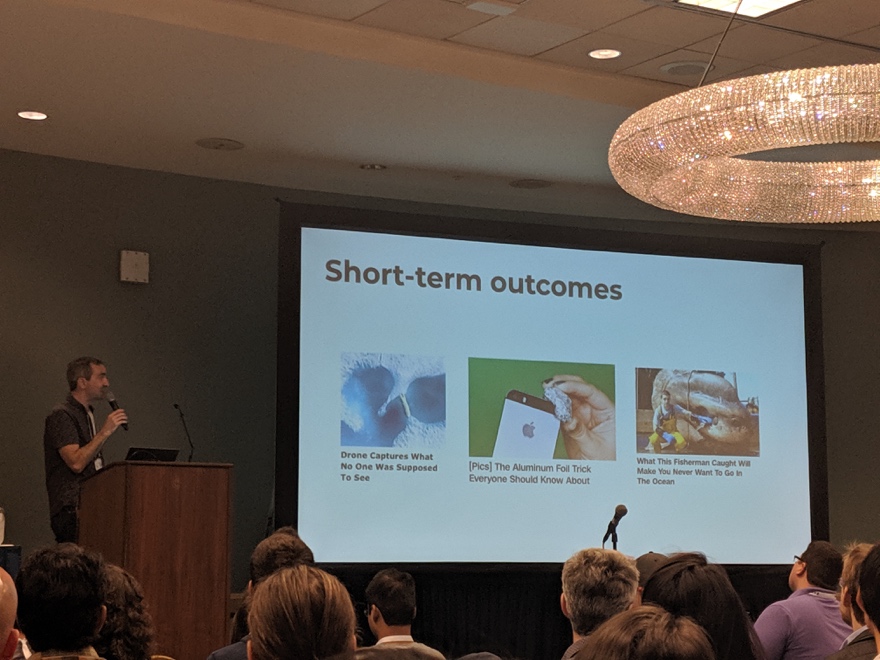
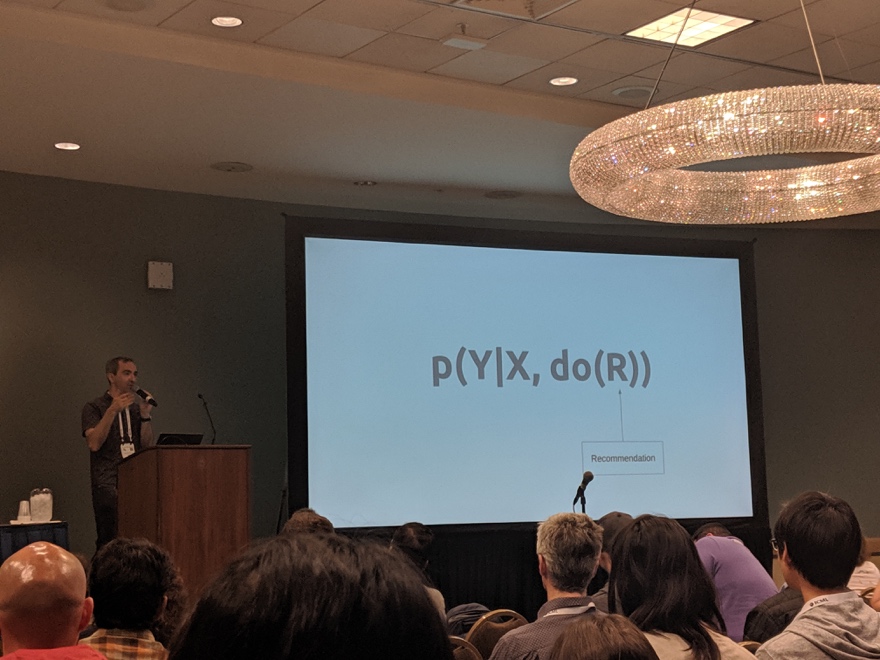
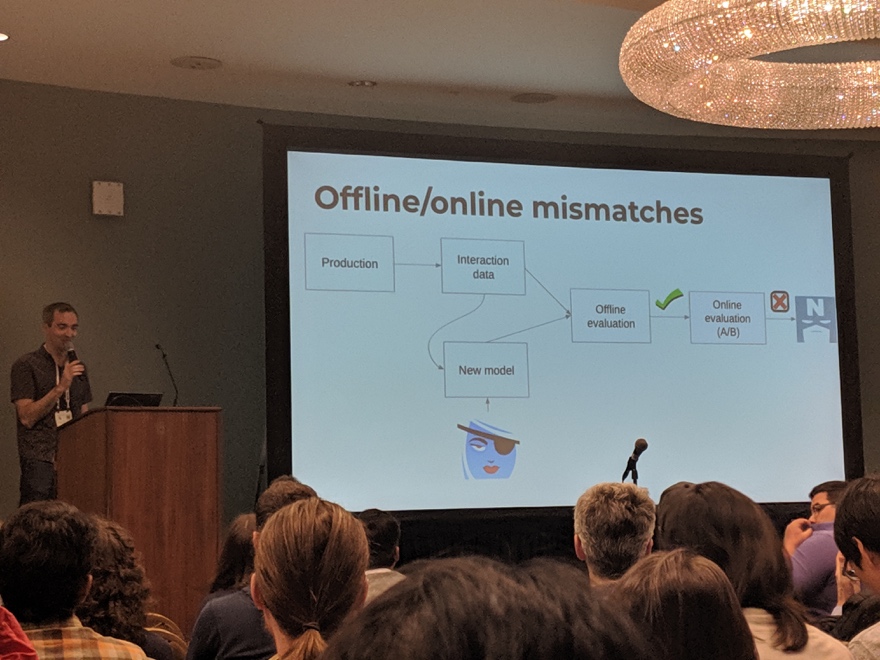
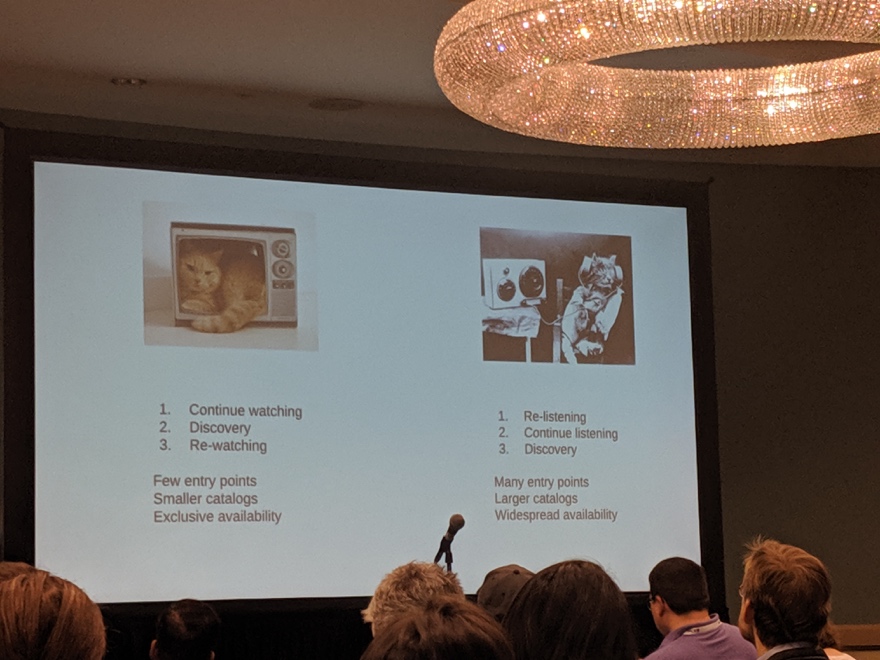
# Workshop: Machine Learning for Music Discovery

Schedule <https://icml.cc/Conferences/2019/Schedule?showEvent=3517>

## From Listening to Watching, A Recommender Systems Perspective @Netflix

Yves Raimond, director of research







* TV v.s. Music
  + Listen
    - * Old favorites
      * New albums by favorite artists
      * Discover new tracks (among ~27M tracks, challenging)
  + Watch
    - * New episodes of favorite show
      * Discover new shows (~1M, smaller space)
      * Old favorites
* Recommend completely new things
  + Rewarding ⇒ personalize for all subscribers
  + Home page, rank
* Diagram
  + Training data -> rec algo -> rec -> feedback -> another iter of training data
  + Feedback = f(recommend)
    - * Good or bad recommendation
      * Explicit / implicit
      * Implicit
        + Engagement
        + Play minutes
      * Explicit
        + Survey
        + Thumb up and down
  + Reward = f(feedback)
    - * From feedback to long term success
* Music: large catalog + weaker feedback
  + Long tail <https://www.springer.com/us/book/9783642132865>
  + Music data graphs circa 2007 <http://musicontology.com>
* From graph to recommendations
  + Knowledge graph ⇒ recommender
* TV: smaller catalog + strong feedback
  + Short tail
  + Past engagement is most predictive of future engagement
  + Collaborative filtering (CF) is the KING
  + Matrix factorization
  + Challenge
* Differential engagement across items
* New items with no engagement
  + CF?
* Dealing with tail
  + Correlation way
    - * p(Y|X): feature predicts outcome, content-based
      * Wavenet
      * <https://scholar.google.com/citations?user=yNNIKJsAAAAJ&hl=en>
  + Back to MF
    - * Feed-forward view
      * U x V => inner product
      * Deeper
  + Add extra info to the matrix
    - * Sequence of past engagement
      * Feature space
      * Baseline sequence + continuous time context
* offline/online mismatch???
* Introduce new or unknown items
  + High or low p(Y|X)
* p(Y|X, do(R))
  + Epsilon-greedy: simple but expensive
  + IPS?: probability weighting, high variance
  + IV?: instrumental va?? hard to scale
  + Others
* Design rewards
  + E[Y|X, do (R)]
  + Maximize expected reward, given recommendation
  + Focus on long term reward
    - * Sequence of actions
      * RL ⇒ delayed reward
  + Optimize for long term

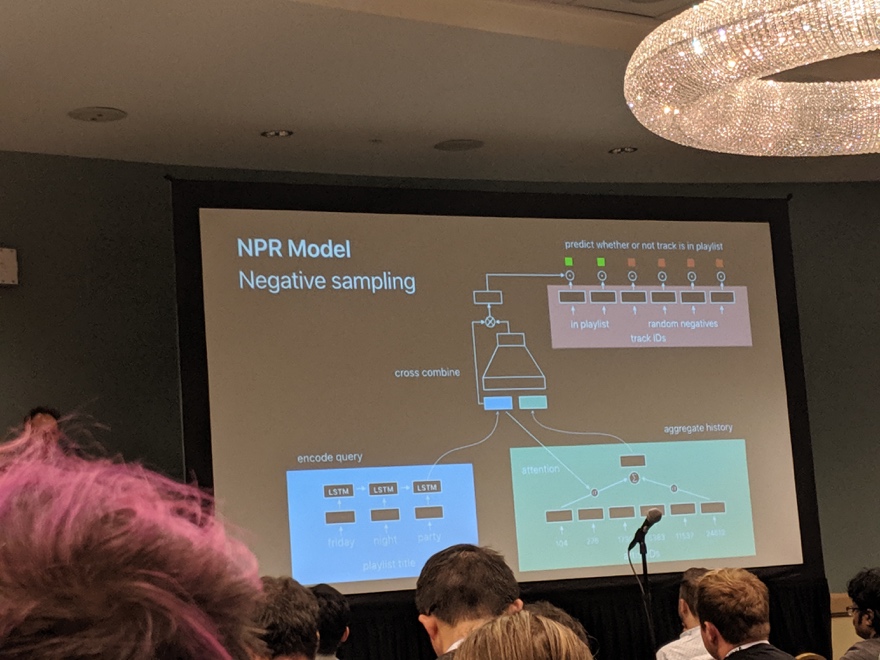
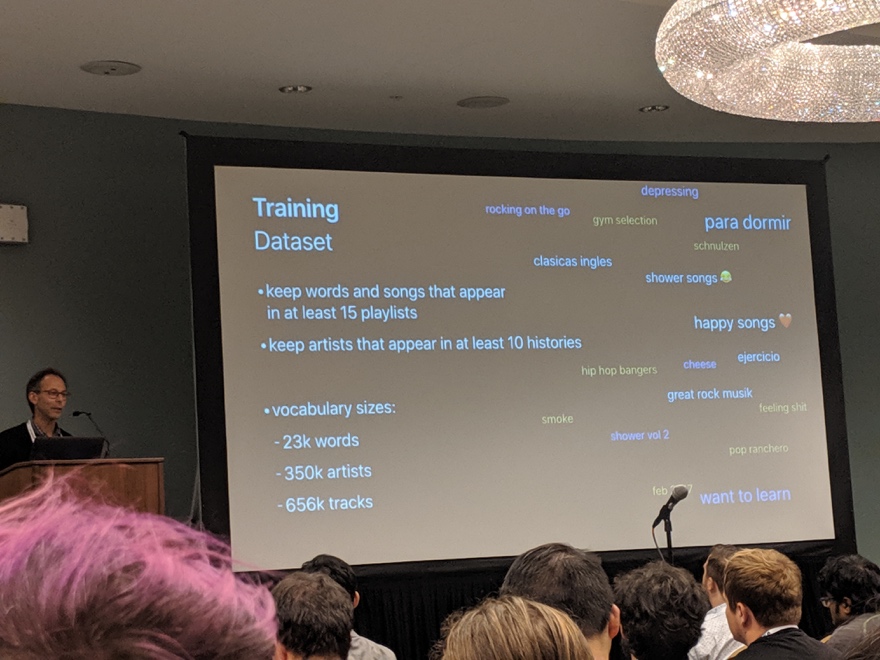
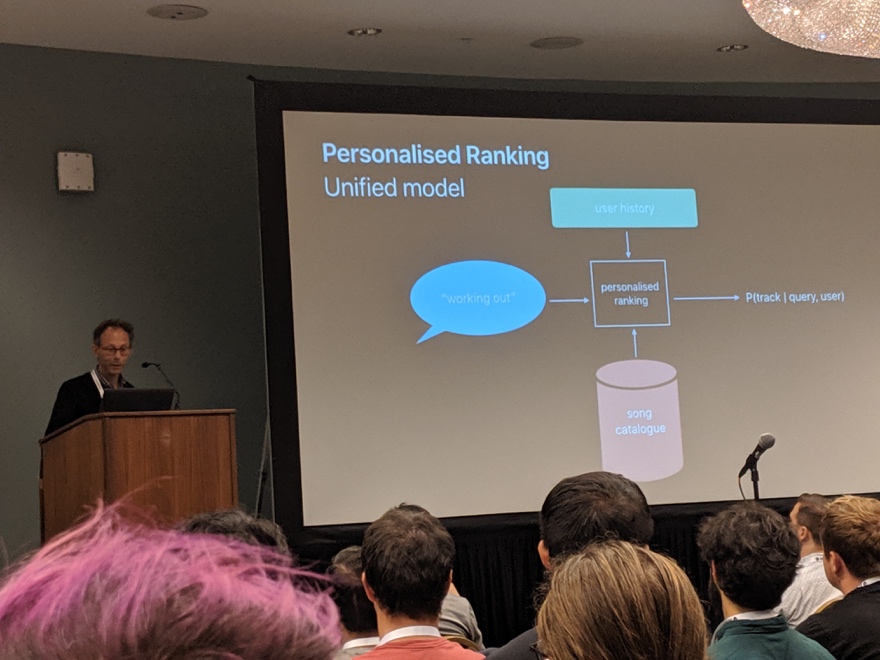
## Characterizing Musical Correlates of Large-Scale Discovery Behavior @ Stanford

Blair Kaneshiro <https://ccrma.stanford.edu/~blairbo/>

* EEG
* audio content recognition services such as Shazam
* Shazam query features
  + Shazam offset study
    - Billboard hit
* <https://www.frontiersin.org/articles/10.3389/fpsyg.2017.00416/full>
* we investigate whether the timing of audio identification queries within a song can be related back to specific musical events.
* We aggregate and analyze a large collection of Shazam query offsets—that moment in a song when a user initiates a query—over a set of massively popular songs
* Histogram of song: number of queries at each time of a song
  + When do people start to query in a song
  + chorus

## NPR: Neural Personalised Ranking for Song Selection @Apple

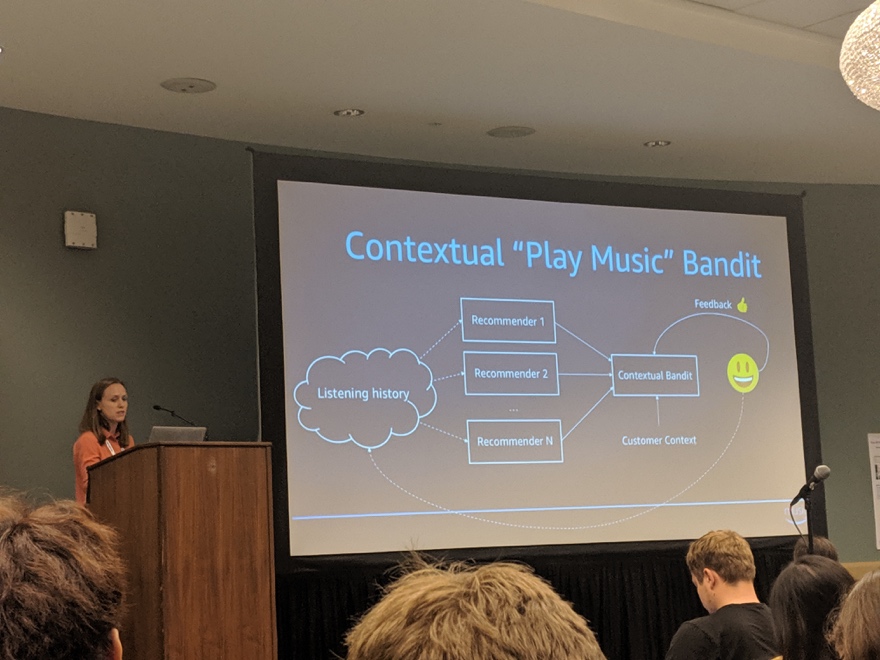
Mark Levy



* Apple music science team
* Personalized ranking
  + Board search => ranking
  + Create a playlist with a title => recommend song list
  + “Play music, alexa”
* Workflow
  + Query => database match + user history taste => rank
* Approach 1: tagging
  + Tagging first
  + Personalization later
* Tagger is not sufficient for ranking
  + Not just display correctly, but also induce listening
* Re-ranking from thousands of songs from ~M songs
  + Users hate all genre in this candidates
  + Language specific
* Use unified model
  + With a lot of data
  + 4.5 million use playlists with titles
  + Basic quality filter
    - Limit playlist per user
  + Use history
    - ~top 200 artists per user
  + Training data
    - Vocabulary
  + Title + query => NPR
    - Softmax
    - Encoded query (word embedding LSTM)
    - Artist id embedding (song pooling??)
    - Embed songs as tensors
  + NPR model with attention
    - How to aggregate over history
    - Weighted sum over time
    - Latent cross
    - Negative sampling (not for all softmax)
  + Optimization
    - Sampled softmax loss
    - Vanilla hyperparameters
      * Gradient descent with fixed learning rate
      * Embedding size 100
  + Setting
    - Attention with 1 or 4 heads
* Demo
  + Music recommendation by search

## Personalization at Amazon Music @Amazon

Kat Ellis



* Alexa, play summer time
  + Training
    - Pairwise ranking model
      * query
      * Maximize prob of ranking positive over negative
* Alexa, play music
  + Popular?
  + Recommendation?
  + Playlist?
* Strategy
  + Listening history ⇒ station recommender
* Understand intent
  + Contextual bandit
  + Customer context
* Embedding models for personalized recommendations
  + Songs for YOU
  + Matrix factorization
    - Minimize Euclidean distance
  + User interest
    - No explicit rating
      * Very sparse
    - Implicit feedback
      * Softmax: user-item interaction
      * Confidence term C(ij)
* Offline evaluation
  + Within-batch sampling
* A/B test
  + Engagement 93%+
* Song embedding
  + t-sNE
* Cold-start
  + Content-based CF ⇒ learn behavioral similarity
  + Large margin ranking loss
* Future work
  + Stability of embedding
  + Presentation bias
  + contextualization

## A Model-Driven Exploration of Accent Within the Amateur Singing Voice

Camille Noufi

* MERI
* Characteristic of voices
  + Fixed: demographic
  + Fleeting: perceived levels, likeability, expressivity
  + Accent: origin
* Timbre / intonation : capture accent
* Kareoke dataset: DAMP by smule
  + <https://ccrma.stanford.edu/damp/>

## What’s Broken in Music Informatics Research? Three Uncomfortable Statements @Adobe

Justin Salamon

* MIR in pitch tracking
  + <http://www.justinsalamon.com/>
  + Track instrument/voice in track
* Applications
  + Query by humming
  + Convert song recog
  + Classify singing style
  + Pattern detection
* 1: melody extraction
  + What is genre, what is melody, what is MUSIC?
  + Arbitrary definition of music related concepts
  + What exactly to extract?
    - ???
  + Definition is determined by the annotation of the dataset
* 2: evaluate algo performance
  + Dataset too small/homogenous
  + Metrics not good
* 3: lean start up
  + <https://medium.com/@blacktar/customer-development-resources-for-startups-7dc2272c94ec>

# Workshop: Learning and Reasoning with Graph-Structured Representations

Schedule <https://graphreason.github.io/schedule.html>

## RL with graph: chemistry and beyond @ Benevolent AI

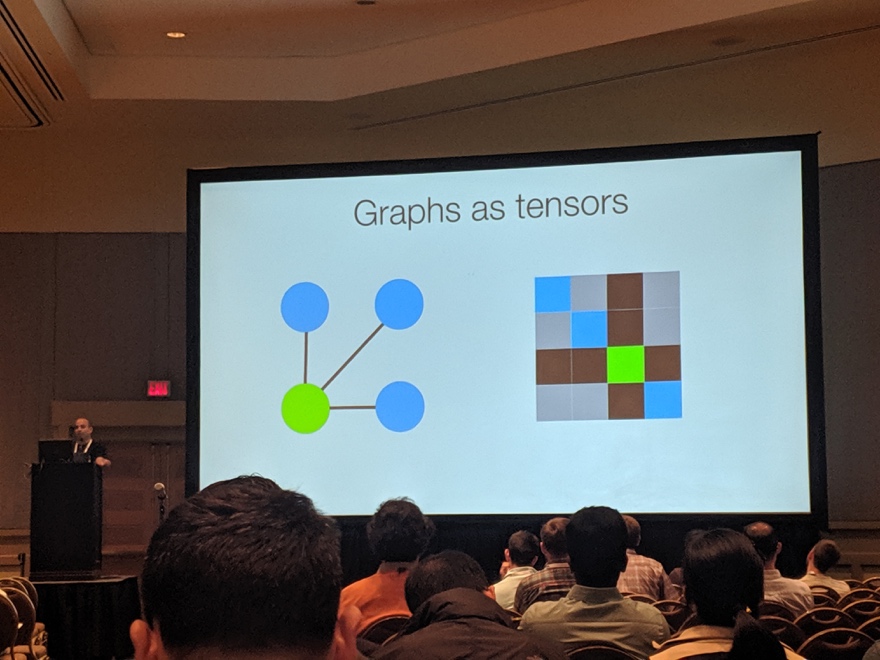
Marwin Segler

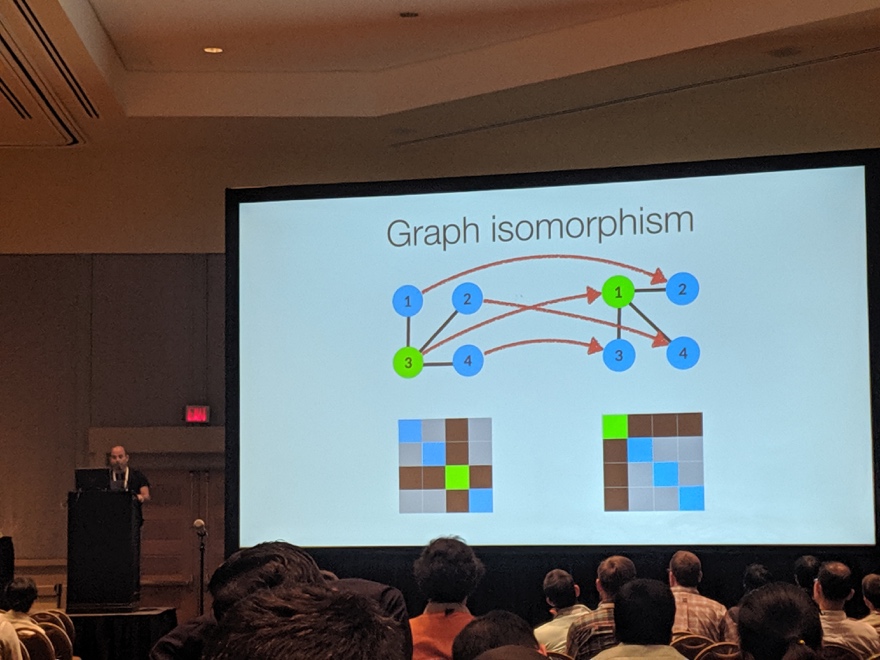
* Graph everywhere
  + chemistry the study of small graphs
  + design->make->test->design
  + ML based design and make
* Generate drugs from data
  + Drug discovery with goal-directed optimization
  + K-L divergence of distribution
* Planning in graph-based environment
  + M = (S,A,T,R)
  + State, action, transition, reward
* How to target a molecule
  + Usually require knowledge, experience, and creativity (similar to chess)
  + Not every molecule is synthesizable YET
* How to make a molecule

## Geometric processing

Yaron Lipman

Weizmann Institute of Science





* Geometric processing
* Graph classification/regression
  + f(GRAPH) => label
* Efficiency of neural net / power
  + Characterize graph geometry with features (HOW)
* Invariant graph networks
  + Graphs not as graph, but as tensors (matrix), adjacency matrix
  + Graph isomorphism (equivalent graph)
    - Matrix => permutation P (dot) A = B
    - Permutation also applies to higher dimension tensor
    - Re-order each dimension
  + Function
    - f(p X) = f(X)
    - Function is the same on X and permuted X
* Graph learning
  + f(MATRIX) => label
  + Linear and point-wise non-linear activation Y = sigma(L(X))
  + Equivariant graph
    - f(pX) = pf(X)
* Invariant graph network (IGN)
  + Implement

# Workshop: Self-supervised learning workshop

Schedule <https://sites.google.com/view/self-supervised-icml2019>

Self-supervised learning is a promising alternative where proxy tasks are developed that allow models and agents to learn without explicit supervision in a way that helps with downstream performance on tasks of interest. One of the major benefits of self-supervised learning is increasing data efficiency: achieving comparable or better performance with less labeled data or fewer environment steps (in Reinforcement learning / Robotics).

# Supervised learning

## Data Shapley: Equitable Valuation of Data for Machine Learning

<https://arxiv.org/abs/1904.02868>

Training + learner + performance

* quantify the value of data in algorithmic predictions and decisions.
  + For example, in healthcare and consumer markets, it has been suggested that individuals **should be compensated for the data that they generate**, but it is not clear what is an equitable valuation for individual data.
  + In this work, we develop a **principled framework to address data valuation in the context of supervised machine learning**.
  + Given a learning algorithm trained on n data points to produce a predictor, we propose **data Shapley** as a metric to quantify the value of each training datum to the predictor performance.

## Feature Grouping as a Stochastic Regularizer for High-Dimensional Structured Data

<https://arxiv.org/abs/1807.11718>

* High dimension + small sample + noise
* Regularization
  + Early stopping
  + L1, l2
  + Group LASSO
  + Stochasticity
  + ReNA: feature grouping Hoyos 2016
    - Fast clustering
    - Signal approximation

## Metrics optimized example weights

<https://arxiv.org/abs/1805.10582>

Slides <https://icml.cc/media/Slides/icml/2019/seasideball(11-11-00)-11-11-25-4392-metric-optimize.pdf>

* Ranking model
  + Train and test are not paired: hinge loss
  + Global and local metrics don’t match
  + precision@3
* MOEW
  + Maximize test metrics
  + <https://openreview.net/pdf?id=SklgHoRqt7>
  + Weighted space

## Improving Model Selection by Employing the Test Data

<http://proceedings.mlr.press/v97/westphal19a.html>

Slides: <https://icml.cc/media/Slides/icml/2019/seasideball(11-11-00)-11-11-30-4393-improving_model.pdf>

* Model selection: we investigate the properties of novel evaluation strategies, namely when the final model is selected based on empirical performances on the test data.
* When in doubt, delay the final model choice to the test data

# Clustering

## Scalable fair clustering @IBM

<https://arxiv.org/abs/1902.03519>

Slides: <https://icml.cc/media/Slides/icml/2019/grandball(11-16-00)-11-16-30-5049-scalable_fair_c.pdf>

* Fair clustering:
* Algo fairness
* Common unsupervised tasks
* Feature engineering
* Balanced cluster (for features)

## DBSCAN++: Towards fast and scalable density clustering

<https://arxiv.org/pdf/1810.13105.pdf>

a simple modification of DBSCAN which only requires computing the **densities for a chosen subset** of points.

* Scalable clustering
* Parametric
  + Minimize loss function with k
* Non-parametric
  + Density
  + Arbitrary shape
  + Groups
    - Density
    - Mode-seeking: meanshift
* DENSITY
  + High concentration data points
  + Euclidean distance
  + **Compare using sk-learn functions**
* DBSCAN
  + Neighbor
  + Radius, core points
  + How it works
* Sampling ⇒ to make it faster
* Greedy k-center
  + NP hard
  + Gradient descent
* Run faster by sampling

## concrete autoencoder

paper: <http://proceedings.mlr.press/v97/balin19a.html>

slides: <https://icml.cc/media/Slides/icml/2019/103(13-11-00)-13-11-20-4961-concrete_autoen.pdf>

* Unsupervised Feature Selection (UFS) is Widely Used in Machine Learning
* Mostly based on L1 L2 regularization
* Identify important features to measure, and then reconstruct other features
* Concrete random variables
  + In: all features
  + Out: top features
* Implementation is just a few lines of code from a standard autoencoder

## Spectral clustering of signed graph

Paper: <https://arxiv.org/abs/1905.06230>

Slides:<https://icml.cc/media/Slides/icml/2019/103(13-11-00)-13-11-30-4963-spectral_cluste.pdf>

* In-cluster: only pos edges
* Out-cluster: only neg edges
* Metrics ⇒ power mean Laplacian
* Use signed graph for clustering

Coreset for Ordered Weighted Clustering

paper: <https://arxiv.org/abs/1903.04351>

slides: <https://icml.cc/media/Slides/icml/2019/103(13-11-00)-13-11-35-4964-coresets_for_or.pdf>

* K-median

## Fair k-Center Clustering for Data Summarization

Paper: <https://arxiv.org/abs/1901.08628>

* K-center problem
* NP hard

## Better k-means ++ @Google research

Paper: <http://proceedings.mlr.press/v97/lattanzi19a.html>

Slides: <https://icml.cc/media/Slides/icml/2019/103(13-11-00)-13-12-00-4966-a_better_k-mean.pdf>

* Local search
  + slow
* K-means
  + Seeding + Lloyd’s algorithm
    - Probably the most celebrated heuristic for k-means is the wellknown Lloyd’s algorithm (Lloyd, 2006). The algorithm is also often called the k-means algorithm. The algorithm usually performs very well in practice, but it does not provide a theoretical approximation guarantee
* Combine both
  + K-means++ seeding with local search
* Experiment with benchmark

## Spectral clustering for fairness constraints

Paper: <https://arxiv.org/pdf/1901.08668.pdf>

Slides: <https://icml.cc/media/Slides/icml/2019/103(13-11-00)-13-12-10-4968-guarantees_for_.pdf>

* Fair clustering (Chierichetti et al. 2017): in every cluster, **each group Vs should be represented with (approximately) the same fraction** as in the whole data set V
* Use fair clustering in our clustering
* NP hard
* Balanced results

## Supervised hierarchical clustering

Paper: <http://proceedings.mlr.press/v97/yadav19a.html>

Slides: <https://icml.cc/media/Slides/icml/2019/103(13-11-00)-13-12-15-4969-supervised_hier.pdf>

* Linkage function (distance)
  + Unknown which one is the best
* Exponential linkage
  + Learnable linkage function
* Weighted edge
* Use errors of hierarchical clustering results to get gradient descent

Training Objective & Algorithm: Jointly Optimizing Dissimilarity & Linkage Function

## Learn to route in similarity graph @Yandex

paper: <https://arxiv.org/abs/1905.10987>

slides: <https://icml.cc/media/Slides/icml/2019/grandball(12-16-00)-12-16-35-4493-learning_to_rou.pdf>

* In KNN search space
* • Recall@1 — a rate of queries for which the actual nearest neighbor is successfully found
* Similarity graph: Edges connect (mostly) nearest neighbors
  + Local search
  + Graph Convolutional Network

## Power k-means clustering

Paper: <http://proceedings.mlr.press/v97/xu19a.html>

Slides: <https://icml.cc/media/Slides/icml/2019/104(13-09-00)-13-09-40-4768-power_k-means_c.pdf>

* lloyd algo
* K-harmonic mean as the criteria, rather than arithmetic mean

## The Wasserstein Transform

paper: <http://proceedings.mlr.press/v97/memoli19a/memoli19a.pdf>

slides: <https://icml.cc/media/Slides/icml/2019/104(13-16-00)-13-16-20-4424-the_wasserstein.pdf>

* Metric space + scale parameter
* for enhancing and denoising datasets defined on general metric spaces
* mean shift also inherits stability under perturbations

## Sequential Facility Location: Approximate Submodularity and Greedy Algorithm

Paper: <http://proceedings.mlr.press/v97/elhamifar19a.html>

Slides: <https://icml.cc/media/Slides/icml/2019/104(13-16-00)-13-16-25-4425-sequential_faci.pdf>

* Subset selection to represent large ground set
* Sequential subset selection ⇒ fast greedy algorithm
* X: source -> y: target
* Procedure learning

## Neural collaborative subspace clustering

paper: <https://arxiv.org/abs/1904.10596>

slides: <https://icml.nips.cc/media/Slides/icml/2019/104(13-16-00)-13-16-30-4426-neural_collabor.pdf>

* Subspace: dimension reduction
* STOA methods
  + Affinity matrix
  + Normalized cut or spectral clustering
* New
  + affinity matrix in batch
  + Train classifier using affinity matrix
* Affinity from classification
* Encoder-decoder for clustering
* Collaborative learning for clustering = classification affinity

## Unsupervised deep learning by neighborhood discovery

paper: <https://arxiv.org/abs/1904.11567>

slides: <https://icml.cc/media/Slides/icml/2019/104(13-16-00)-13-16-35-4427-unsupervised_de.pdf>

* Clustering analysis
* Consistent neighborhood, similarity
* Use entropy to learn class consistency

# Autoregressive energy machine ARNN @deepmind

Paper: <https://arxiv.org/abs/1904.05626>

* Model joint distribution of random variables at high dimension
* Deep generative model
* Importance sampling not work well for high dimension
* Auto-regressive nn
  + Output depends on independent

## Noise2Self: Blind Denoising by Self-Supervision

Paper: <https://arxiv.org/abs/1901.11365>

Slides: <https://icml.cc/media/Slides/icml/2019/104(13-16-00)-13-17-05-4430-noise2self_bli.pdf>

* Biomedical data (noisy)
* Protein imaging (s/n: 1:10)
* Noise removal with supervision
* Self-supervision
  + f(x)-x?
  + expected
* Conditional expectation of noise pixels
* With CNN training
* De-noise image

## Learning Dependency Structures for Weak Supervision Models

Paper: <https://arxiv.org/abs/1903.05844>

Slides: <https://icml.cc/media/Slides/icml/2019/104(13-16-00)-13-17-10-4431-learning_depend.pdf>

* Snorkel: widely used in industry
  + Deal with noisily labeled data
  + Model the label functions’ behavior to denoise
  + Probabilistic labels
* Probabilistic generative training data
* <https://dawn.cs.stanford.edu/2017/12/01/snorkel-programming/>

# Applications

## Exploiting Worker Correlation for Label Aggregation in Crowdsourcing @facebook

Paper: <http://proceedings.mlr.press/v97/li19i.html>

* inferring consensus aggregation of collected annotations is a core crowdsourcing task, with the simplest technique being majority voting.
* **Numerous probabilistic models** have emerged that parameterise worker reliability to improve consensus accuracy (David & Skene, 1979; Whitehill et al., 2009; Kim & Ghahramani, 2012).
* Multiple label for the same item, and aggregate labels on the same item, to infer true label
  + Based on the observed label
* Also with missing values
* Bayesian classifier
  + Joint distribution of worker labels + true labels ⇒ marginal prob
  + **True label is first generated, and then item labels are generated (generative model)**
* Worker labels are independent iBCC / dependent BCC
  + Conditional prob
* Tensor rank decomposition

## Efficient Amortised Bayesian Inference for Hierarchical and Nonlinear Dynamical Systems

Paper: <https://arxiv.org/abs/1905.12090>

Slides: <https://icml.cc/media/Slides/icml/2019/201(12-11-00)-12-11-20-4906-efficient_amort.pdf>

* Synthetic biology / generating pattern
* **Genetic circuit in** biological function E-coli molecular engineering
* Simulate differential behaviors

## A Multitask Multiple Kernel Learning Algorithm for Survival Analysis with Application to Cancer Biology

Paper: <http://proceedings.mlr.press/v97/dereli19a/dereli19a.pdf>

Slides: <https://icml.cc/media/Slides/icml/2019/201(12-11-00)-12-11-25-4907-a_multitask_mul.pdf>

* A multitask multiple kernel learning algorithm
* Use chip sequencing info to identify cancer types
* Kernel learning, survival kernel
* Design new algo with some theoretical

## Fast and Flexible Inference of Joint Distributions from their Marginals

Paper: <http://proceedings.mlr.press/v97/frogner19a/frogner19a.pdf>

Slides: <https://icml.cc/media/Slides/icml/2019/201(12-11-00)-12-11-30-4908-fast_and_flexib.pdf>

* Learning: infer 2 random variables with joint dist
* conditional dist
* From known to infer unknown

## Cognitive Model Priors for Predicting Human Decisions

Paper <https://arxiv.org/abs/1905.09397>

Slides <https://icml.cc/media/Slides/icml/2019/201(12-11-00)-12-11-35-4909-cognitive_model.pdf>

* Behavioral science
* ML
  + Flexibility
  + Hard to generalize
* Cognitive model prior ⇒ better generalization
  + Input ⇒ behavior
  + Synthetic data
* Train NN
* BEAST
  + Black box model
* Training size ⇒ variation

## Conditioning by adaptive sampling for robust design

Paper: <https://arxiv.org/abs/1901.10060>

Slides: <https://icml.cc/media/Slides/icml/2019/201(12-11-00)-12-11-40-4910-conditioning_by.pdf>

* Design protein seq for function
* Existing
  + Known sequence ⇒ function (ML)
* New
  + From function, design sequence
* **Model-based optimization**
  + Adaptive sampling
  + MC sampling and anneal
  + Timestamps sample sequence
* Pathological oracles
* Deal with small training size
* Minimize KL divergence
* Testing is different
  + Ground truth design: GP mean function
    - How to train a GP
    - And then sample from GP
  + No good set of hold-out
* Compare with bayesian opto

## Direct Uncertainty Prediction for Medical Second Opinions @google brain

Paper: <https://arxiv.org/abs/1807.01771>

Slides: <https://icml.cc/media/Slides/icml/2019/201(12-11-00)-12-12-00-4911-direct_uncertai.pdf>

* Second opinion among doctors
  + Source of disagreement from patient’s profile
* **Flag patients with 2nd opinions**
* Approach
  + Uncertainty via classification
  + Direct uncertainty prediction DUP
* Hidden features seen by doc, not by photos

## Dynamic Measurement Scheduling for Event Forecasting Using Deep RL

Paper: <https://arxiv.org/abs/1901.09699>

Slides: <https://icml.cc/media/Slides/icml/2019/201(12-11-00)-12-12-05-4912-dynamic_measure.pdf>

Dynamic treatment scheduling

Patients outcome -- measurement

Uniform policy from

DQN-RL, a lot of action space

* RL framework for cost-sensitive scheduling of measurements in time-series
* Scalable to large number of measurements
* Promising results in a real-world ICU dataset (MIMIC3)

## DeepNose: Using artificial neural networks to represent the space of odorants @ cold spring harbor

Paper: <http://proceedings.mlr.press/v97/tran19b/tran19b.pdf>

<https://icml.cc/media/Slides/icml/2019/201(12-11-00)-12-12-15-4914-deepnose_using.pdf>

* Odorant receptor combination
  + 3D molecular filter
  + Train with neural net
* Autoencoder
* Classifier using encoder’s features
* Dimension reduction for odor cloud

## Doubly Robust Joint Learning for Recommendation on Data Missing Not at Random

Paper: <http://proceedings.mlr.press/v97/wang19n/wang19n.pdf>

Slides: <https://icml.cc/media/Slides/icml/2019/201(13-09-00)-13-09-00-4916-doubly_robust_j.pdf>

* Missing Ratings: Missing Not at Random
* Producer / user
* Error imputation

## Linear-Complexity Data-Parallel Earth Mover’s Distance Approximations @IBM

Paper: <https://arxiv.org/abs/1812.02091>

Slides: <https://icml.cc/media/Slides/icml/2019/201(13-09-00)-13-09-20-4917-linear-complexi.pdf>

* Network problem
* Search accuracy/complexity
  + GPU friendly/optimality
* Introduce some new Relaxed in-flow constraints

## Model Comparison for Semantic Grouping

Paper: <https://arxiv.org/abs/1904.13323>

Slides: <https://icml.cc/media/Slides/icml/2019/201(13-09-00)-13-09-25-4918-model_compariso.pdf>

* Semantic group
* Bag of words embedding
* Hypothesis testing: whether 2 sentences come from the same semantic group
  + Bayes Factor - Integrates out Parameters

## Rank aware factorization machines @Tencent

Paper: <https://arxiv.org/abs/1905.07570>

Slides: <https://icml.cc/media/Slides/icml/2019/201(13-09-00)-13-09-30-4919-rafm_rank-awar.pdf>

* Embedding with weight
* FM: inner product

## CAB: Continuous Adaptive Blending Estimator for Policy Evaluation and Learning

Paper: <https://arxiv.org/abs/1811.02672>

Slides: <https://icml.cc/media/Slides/icml/2019/201(13-09-00)-13-09-35-4920-cab_continuous.pdf>

* Online policy learning and evaluation
* Online A/B testing
* Interpolated Counterfactual Estimator Family

# Time series

## Generative Adversarial User Model for Reinforcement Learning Based Recommendation System

Paper <https://arxiv.org/abs/1812.10613>

Slides <https://icml.cc/media/Slides/icml/2019/201(11-14-00)-11-14-25-4831-generative_adve.pdf>

* Generative for RL and RS
* In RS, experience is updated from historical observation
* Challenges
  + User is the environment, not a lot of interactions
  + Reward function is unknown
* Generative simulator
* **Use GAN**
* Sequential information: LSTM / position weight (attention)
* Cascading Q network
* CTR

## Deep Factors with Gaussian Processes for Forecasting @Amazon AWS

<https://arxiv.org/pdf/1812.00098.pdf>

Slides: <https://icml.cc/media/Slides/icml/2019/201(11-14-00)-11-14-35-4833-deep_factors_fo.pdf>

* Classical time series models fail to fit data well and to scale to large problems, but succeed at providing uncertainty estimates. The converse is true for deep neural networks.
  + Classical time series methods, such as **Autoregressive Integrated Moving Average (ARIMA) [3], exponential smoothing [4] and general Bayesian time series [5]**, excel at modeling the complex dynamics of individual time series of sufficiently long history.
  + These methods are **local, that is, they learn one model per time series**. As a consequence, they cannot effectively extract information across multiple time series. These classical methods also have challenges with **cold-start problems**, where more time series are added or removed over time.
* Deep neural networks (DNNs), in particular, r**ecurrent neural networks (RNNs), such as LSTMs** [6] have been successful in time series forecasting [7, 8]. DNNs are generally effective at extracting patterns across multiple time series. Without a combination with probabilistic methods, such as variational dropout [9] and deep Kalman filters [10], DNNs can be prone to overfitting and have challenges in modeling uncertainty [11].
* we propose a hybrid model that incorporates the benefits of both approaches. Our new method is data-driven and scalable via a latent, global, deep component. It also **handles uncertainty through a local classical Gaussian Process model**. Our experiments demonstrate that our method obtains higher accuracy than state-of-the-art methods.
* Model all time series jointly and globally (works better, complex model, impossible to add more) global/local model

## Weakly-Supervised Temporal Localization via Occurrence Count Learning

Paper <http://users.cs.cf.ac.uk/SchroeterJ1/docs/Weakly-Supervised%20Temporal%20Localization%20via%20Occurrence%20Count%20Learning.pdf>

Github

<https://github.com/SchroeterJulien/ICML-2019-Weakly-Supervised-Temporal-Localization-via-Occurrence-Count-Learning>

Slides: <https://icml.cc/media/Slides/icml/2019/201(11-14-00)-11-14-40-4834-weakly-supervis.pdf>

Obj

* Weakening the annotation requirement
* **Occurrence count**
* Training / inference
* Weakly-supervised
* Counting model
  + Localization
* Loss
  + Compare loss by K-L divergence of the count
* Sound signal ⇒ fourier transform ⇒ CNN  convolution representation ⇒ LSTM

## Imputing missing events in continuous time event series

Paper: <https://arxiv.org/abs/1612.09328>

Slides: <https://icml.cc/media/Slides/icml/2019/201(11-14-00)-11-15-05-4836-imputing_missin.pdf>

* Incomplete data
* when/where ?
* Impute the missing event
  + neural hawkes process
* Game logs retrospective
* Sequential monte carlo

# ML System

## Fault tolerance in iterative convergent ML @Petuum

<https://arxiv.org/abs/1810.07354>

* Distributed ML for model training
* Parameter server architecture (worker and namenode)
  + Unreliable environment at challenge
  + Node failure
  + Slow network
* Tradition: strict consistency model ⇒ inefficient for ML
* Better for ML: Relaxed consistency model
  + Bad: designed for SGD specific algo
* ML training is **self-correcting** in general
  + Unreliable env ⇒ parameter perturbation ⇒ convergence with cost
* Assumption
  + Iterative algo
  + Faults as perturbations
  + Convergence
* Fault-tolerance from priority checkpoints
* System fault can be captured as perturbation

## Static automatic batching in tensorflow @ Google brain

Paper <http://proceedings.mlr.press/v97/agarwal19a/agarwal19a.pdf>

Slides <https://icml.cc/media/Slides/icml/2019/grandball(11-16-00)-11-17-00-5052-static_automati.pdf>

* Tf nodes graph (pytorch though)
* Vectorization challenges

# Reinforcement learning

## Control Regularization for Reduced Variance Reinforcement Learning

Paper: <https://arxiv.org/abs/1905.05380>

Slides: <https://icml.cc/media/Slides/icml/2019/hallb(12-14-00)-12-14-20-4610-control_regular.pdf>

* Optimal policy, distribution of state/action
* High variance for RL during learning
  + Use some control prior for policy
* Control Regularization helps by providing:
  + Reduced variance
  + Higher rewards
  + Faster learning
  + Potential safety guarantees

## On the Generalization Gap in Reinforcement Learning Reparameterizable @salesforce research

Paper: <http://proceedings.mlr.press/v97/wang19o.html>

Slides: <https://icml.cc/media/Slides/icml/2019/hallb(12-14-00)-12-14-25-4611-on_the_generali.pdf>

* Intrinsic gap: RL classic learning theory
* Extrinsic gap: train/test diff

## RL in internet congestion control

Paper: <https://arxiv.org/pdf/1810.03259.pdf>

Slides: <https://icml.cc/media/Slides/icml/2019/hallb(12-14-00)-12-14-35-4613-a_deep_reinforc.pdf>

* Latency measurement
* New congestion control timeline
  + Reward-based architecture: PCC
* Replace gradient descent with RL network

## model based active exploration MAX @ NNAISENSE

paper: <https://arxiv.org/abs/1810.12162>

* E-e tradeoff
* Pure exploration without reward
* task-agnostic  ⇒ similar to unsupervised pre-training followed by fine-tuning
* Bayesian perspective of the exploration
  + Experience changes information gain (posterior info)
* Active.reactive exploration
  + Reactive: provoked by env
  + Active: not provoked by env, science test hypothesis
* Explored area v.s. Unexplored area (model disagree)
* Model-based ⇒ prediction of policy distribution (s,a) => r
* Convert exploration to BO

## inverse RL with ranking

paper: <https://arxiv.org/abs/1904.06387>

slides: <https://icml.cc/media/Slides/icml/2019/hallb(12-14-00)-12-15-00-4615-extrapolating_b.pdf>

Good policy from bad demonstrations

* Use ranked demonstration, allow for extrapolation
* T-REX
* Used to study swarm and multi-agent learning

## Adversarially Learned Representations for Information Obfuscation and Inference

paper: <http://proceedings.mlr.press/v97/bertran19a.html>

slides: <https://icml.cc/media/Slides/icml/2019/grandball(12-16-00)-12-16-00-4489-adversarially_l.pdf>

* Why user decides to share data
* Learn space-preserving representations that obfuscate sensitive information while preserving utility
* Filter image to obscure sensitive information
  + Gender
  + Emotion
* KL divergence

## adaptive neural tree @microsoft

paper: <https://arxiv.org/abs/1807.06699>

slides: <https://icml.cc/media/Slides/icml/2019/grandball(12-16-00)-12-16-20-4490-adaptive_neural.pdf>

* Deep neural net + decision tree
  + DNN: feature learning, scalable, architecture hand designed
  + DT: hand designed features, architecture learned
* ANT:
  + DTs which uses NNs in every path and routing decisions.
  + DT-like architecture growth using SGD
* Growth by SGD
* Unsupervised hierachrical clustering with ANT??!!!

Invited talk

## Learning @ cognitive science

* AI learning v.s. kid learning
* <https://icml.cc/Conferences/2019/Schedule?showEvent=4334>
* Generalization with little data
* MESS: model-building exploratory, social learning system
  + for DARPA grant
  + Abstract causal model from stats evidence
  + Active learning through exploratory play
  + Social learning by imitation
  + Development => e-e trade-off

Child development cognitive psychology

* Psychological Bulletin 2012 gopnik and wellman
* <https://www.ncbi.nlm.nih.gov/pubmed/22582739>

## Best paper: challenge assumption in unsupervised learning representation

**Francesco Locatello, Stefan Bauer, Mario Lucic, Gunnar Rätsch, Sylvain Gelly, Bernhard Schölkopf, Olivier Bachem**

Paper: <https://arxiv.org/abs/1811.12359>

Google blog <https://ai.googleblog.com/2019/04/evaluating-unsupervised-learning-of.html>

* Unsupervised representation learning
  + understand high-dimensional data, and to distill that knowledge into useful representations in an unsupervised manner, remains a key challenge in deep learning.
  + One approach is disentangled representations: capture the independent features of a given scene in such a way that **if one feature changes, the others remain unaffected**
  + We do not find any empirical evidence that the **considered models can be used to reliably learn disentangled representations in an unsupervised way,**

# Expo

## Criteo AI Lab

* Ad research
* DSP
* <https://ailab.criteo.com/hyper-parameter-optimization-algorithms-a-short-review/>

## SigOpt

* Bayesian Opto API
* ~ 6 month trial, ~ 10k per month
* Two Sigma Blog: [Why Two Sigma is Using SigOpt for Automated Parameter Tuning](https://www.twosigma.com/news/article/why-two-sigma-is-using-sigopt-for-automated-parameter-tuning/)
* NVIDIA Blog: [Deep Learning Hyperparameter Optimization with Competing Objectives](https://devblogs.nvidia.com/sigopt-deep-learning-hyperparameter-optimization/)
* NVIDIA Blog: [Optimizing End-to-End Memory Networks Using SigOpt and GPUs](https://devblogs.nvidia.com/optimizing-end-to-end-memory-networks-using-sigopt-gpus/)
* AWS Blog: [Fast CNN Tuning with AWS GPU Instances and SigOpt](https://aws.amazon.com/blogs/machine-learning/fast-cnn-tuning-with-aws-gpu-instances-and-sigopt/)
* ICML Paper: [Evaluation System for a Bayesian Optimization Service](https://arxiv.org/abs/1605.06170)
* ICML Paper: [A Stratified Analysis of Bayesian Optimization Methods](https://arxiv.org/abs/1603.09441)

## Yandex

CatBoost

<https://tech.yandex.com/catboost/>

* Python API
* Compatible with Java
* Internal categorical feature transformation