It Takes a Village to Raise a Machine Learning Model

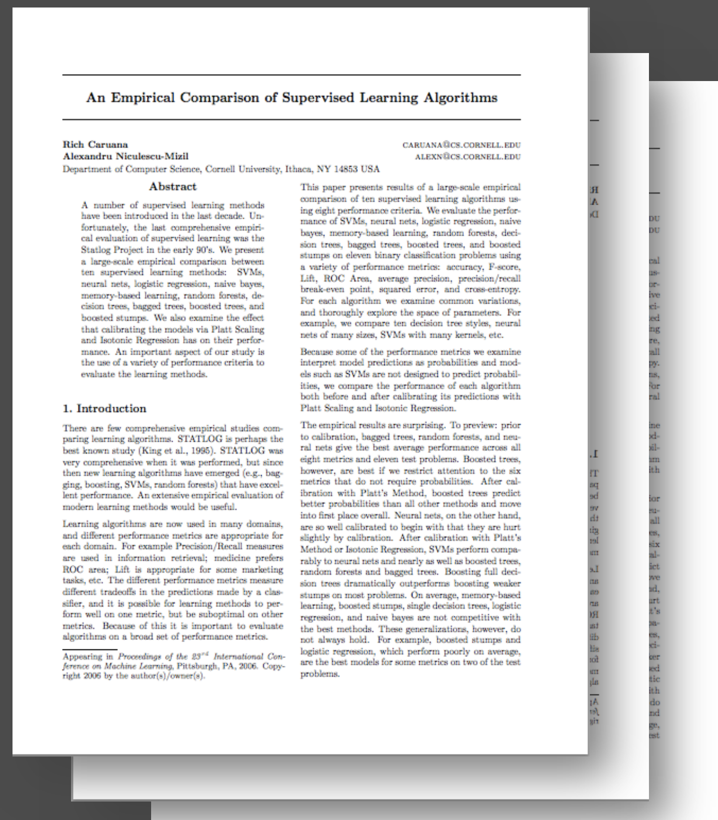
Lucian Lita
@datariver

intuit

It Takes a Village to Raise a Machine Learning Model

Lucian Lita
@datariver

Algorithms



An Empirical Comparison of Supervised Learning Algorithms

Rich Caruana CARUANA@CS.CORNELL.EDU
Alexandru Niculescu-Mizil ALEXN@CS.CORNELL.EDU
Department of Computer Science, Cornell University, Ithaca, NY 14853 USA

Abstract

A number of supervised learning methods have been introduced in the last decade. Unfortunately, the last comprehensive empirical evaluation of supervised learning was the Statlog Project in the early 90's. We present a large-scale empirical comparison between ten supervised learning methods: SVMs, neural nets, logistic regression, naive bayes, memory-based learning, random forests, decision trees, bagged trees, boosted trees, and boosted stumps. We also examine the effect that calibrating the models via Platt Scaling and Isotonic Regression has on their performance. An important aspect of our study is the use of a variety of performance criteria to evaluate the learning methods.

1. Introduction

There are few comprehensive empirical studies comparing learning algorithms. STATLOG is perhaps the best known study (King et al., 1995). STATLOG was very comprehensive when it was performed, but since then new learning algorithms have emerged (e.g., bagging, boosting, SVMs, random forests) that have excellent performance. An extensive empirical evaluation of modern learning methods would be useful.

Learning algorithms are now used in many domains, and different performance metrics are appropriate for each domain. For example Precision/Recall measures are used in information retrieval; medicine prefers ROC area; Lift is appropriate for some marketing tasks, etc. The different performance metrics measure different tradeoffs in the predictions made by a classifier, and it is possible for learning methods to perform well on one metric, but be suboptimal on other metrics. Because of this it is important to evaluate algorithms on a broad set of performance metrics.

This paper presents results of a large-scale empirical comparison of ten supervised learning algorithms using eight performance criteria. We evaluate the performance of SVMs, neural nets, logistic regression, naive bayes, memory-based learning, random forests, decision trees, bagged trees, boosted trees, and boosted stumps on eleven binary classification problems using a variety of performance metrics: accuracy, F-score, Lift, ROC Area, average precision, precision/recall break-even point, squared error, and cross-entropy. For each algorithm we examine common variations, and thoroughly explore the space of parameters. For example, we compare ten decision tree styles, neural nets of many sizes, SVMs with many kernels, etc.

Because some of the performance metrics we examine interpret model predictions as probabilities and models such as SVMs are not designed to predict probabilities, we compare the performance of each algorithm both before and after calibrating its predictions with Platt Scaling and Isotonic Regression.

The empirical results are surprising. To preview: prior to calibration, bagged trees, random forests, and neural nets give the best average performance across all eight metrics and eleven test problems. Boosted trees, however, are best if we restrict attention to the six metrics that do not require probabilities. After calibration with Platt's Method, boosted trees predict better probabilities than all other methods and move into first place overall. Neural nets, on the other hand, are so well calibrated to begin with that they are hurt slightly by calibration. After calibration with Platt's Method or Isotonic Regression, SVMs perform comparably to neural nets and nearly as well as boosted trees, random forests and bagged trees. Boosting full decision trees dramatically outperforms boosting weaker stumps on most problems. On average, memory-based learning, boosted stumps, single decision trees, logistic regression, and naive bayes are not competitive with the best methods. These generalizations, however, do not always hold. For example, boosted stumps and logistic regression, which perform poorly on average, are the best models for some metrics on two of the test problems.

Appearing in *Proceedings of the 23rd International Conference on Machine Learning*, Pittsburgh, PA, 2006.

Data

Big Data Sheep @bigdatasheep • 5yr

more data is better than complex algorithms #BigData

10 14

Big Data Sheep @bigdatasheep • 4yr

more **clean data** is better than more data #BigData

10 14

Big Data Sheep @bigdatasheep • 3yr

more **labeled data** is better than more data #BigData

10 14

Big Data Sheep @bigdatasheep • 2yr

more **smart data** is better than purple data #BigData

10 14

**inflated historical depiction

Data

intuit

Next Frontier: well designed software architectures

Personalization, experimentation, anomaly detection, fraud detection …

Battle Plan

Personalization deep dive
sw architecture flavor

Anomaly detection quick peek

Music streaming, advertising, medical informatics brief stories

intuit mint

intuit TurboTax

intuit QuickBooks®

Product as is.
No customization.

Reasonable coverage.
Segmentation.

… x 1

… x 1

… x 1

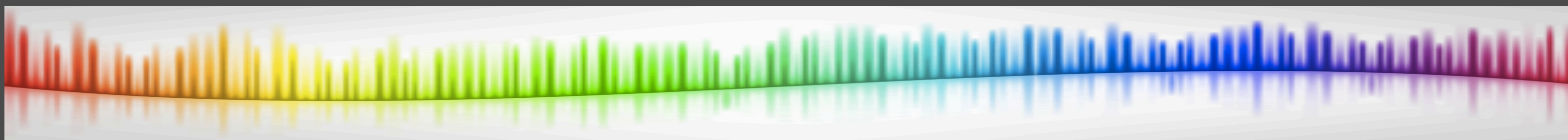
… x 1

Reasonable coverage.
Personalization.

Childhood. Approaches.

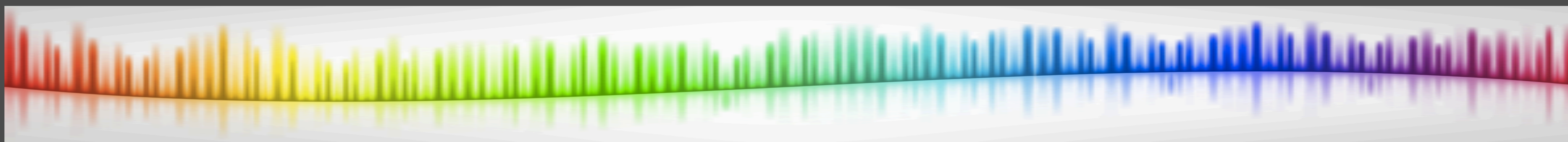
Broad

Deep



Push-button

Push-scientist



App

App

API

delivery

storage

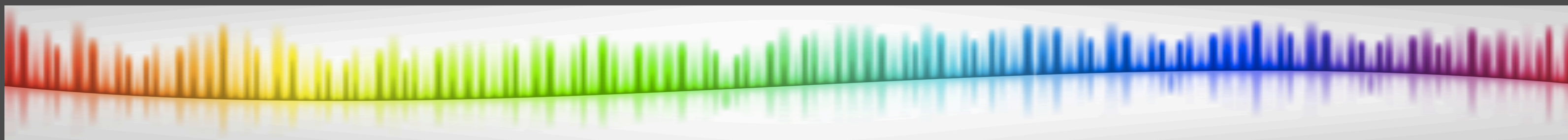
Optimization

-- ML algorithms

-- data: more, better, smarter

-- features, selection

Push-button　　　　Push-scientist

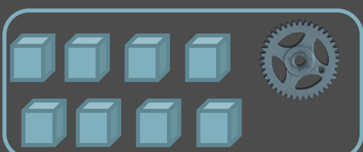


App

API

delivery

storage



Scale & Automation
-- model build
-- model deploy
-- single instrumentation

App

API

delivery

storage

Optimization
-- ML algorithms
-- data: more, better, smarter
-- features, selection

Push-scientist

Invest in ML; start with a thin system

How much effort put into Platform & Automation?

(A) best you can do in x weeks
(B) one step above prototype
(C) enough baling wire & duct tape to support a first use case

Push-button

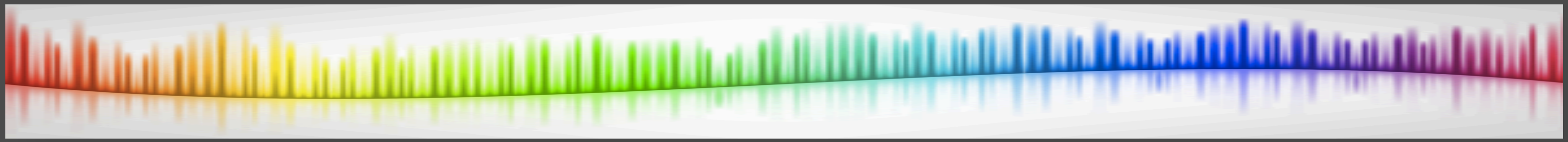
Invest in scale & automation; basic ML

How much effort put into ML?

(A) best generic model setup in y weeks?
(B) noticeably better than random?
(C) pack enough punch to be visible, but not more

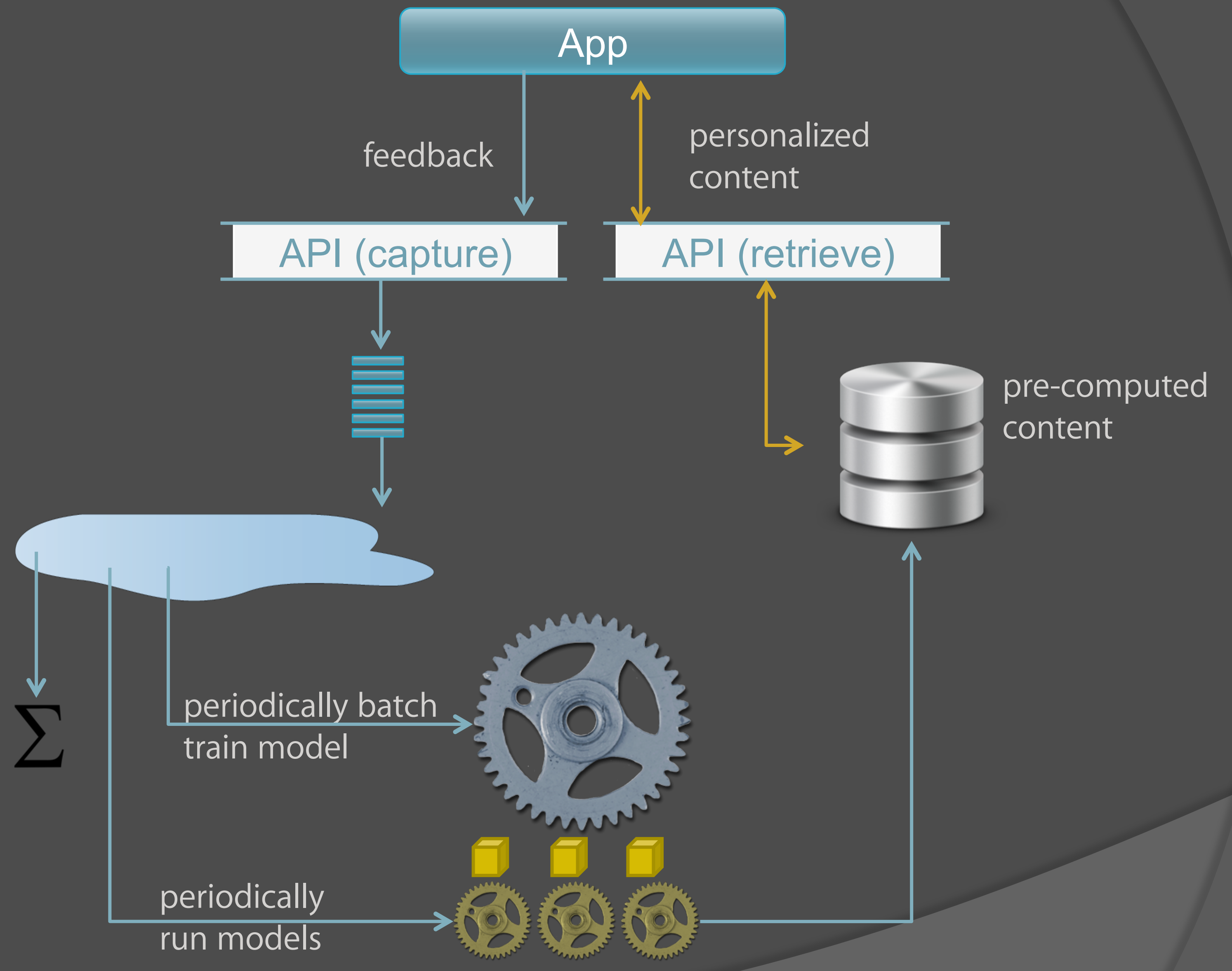
Push-button

Push-scientist

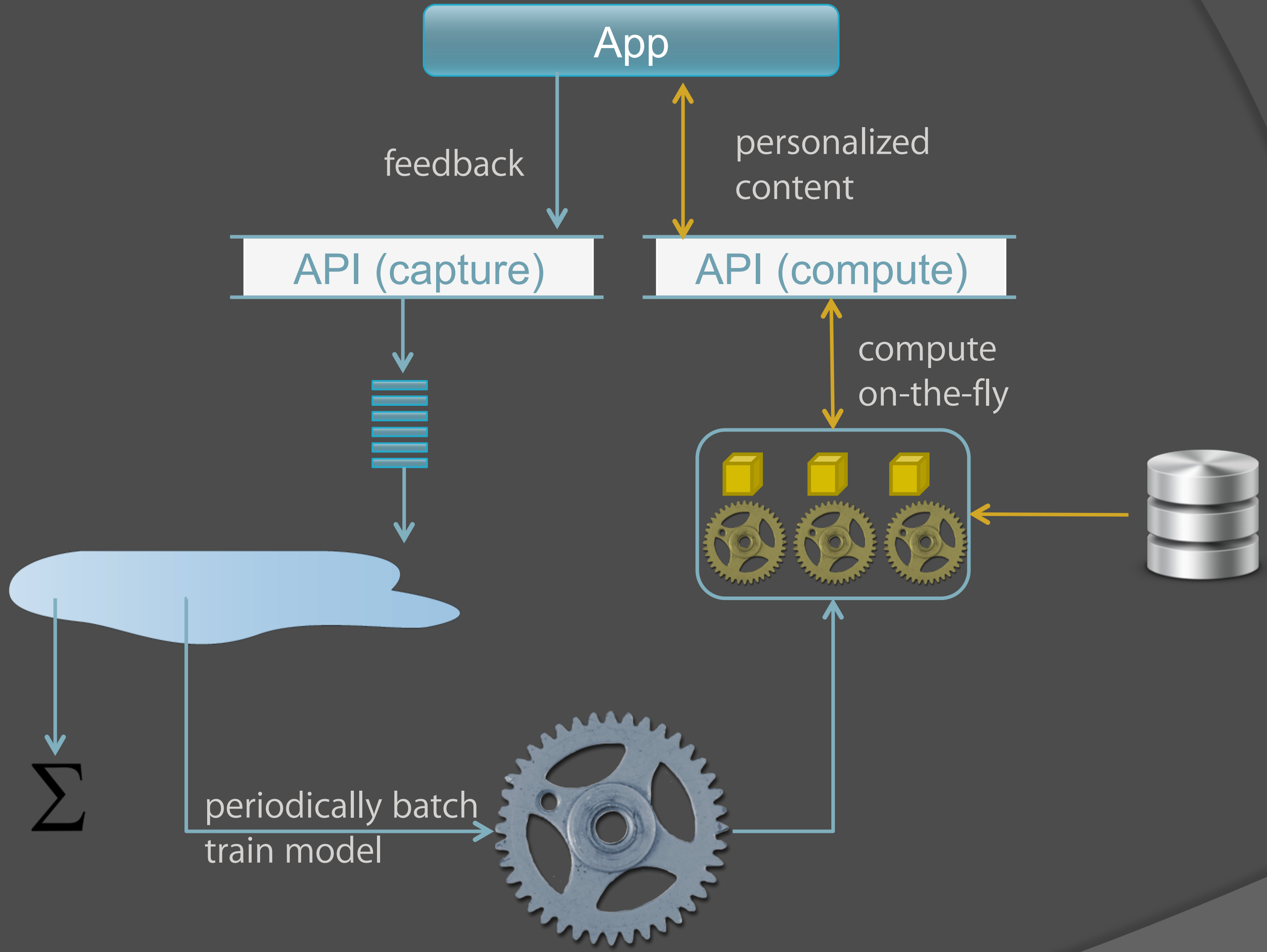


Adolescence. Platform Patterns.

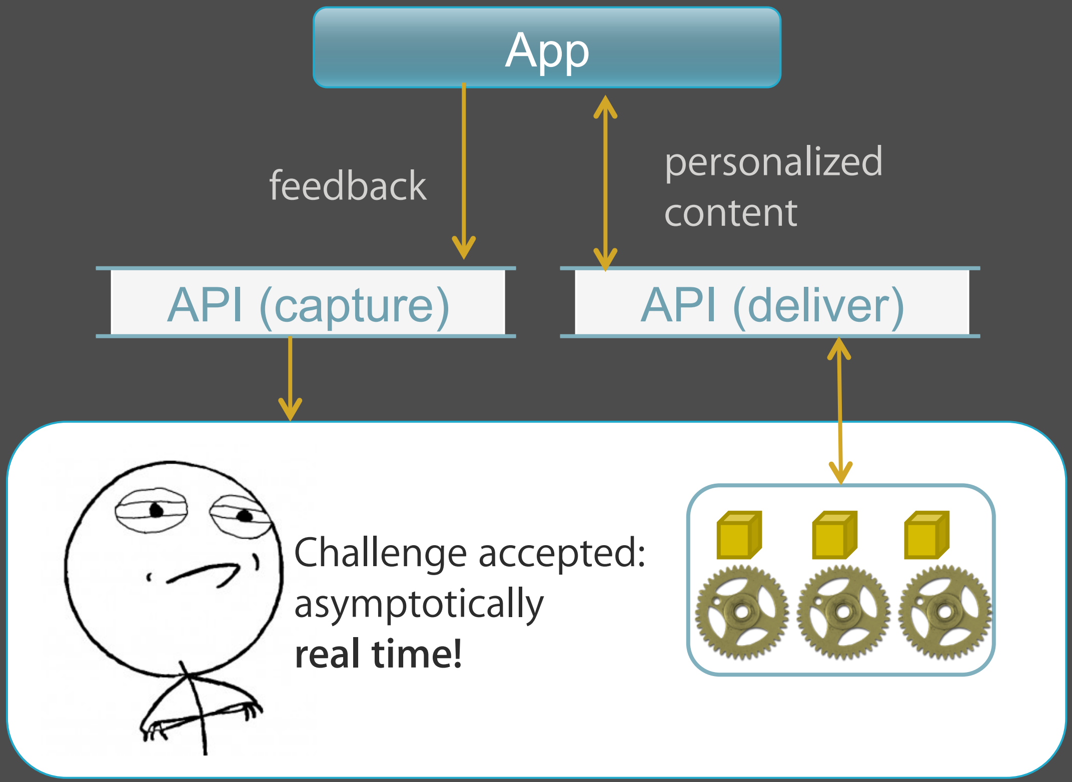
(A) Stored


App
feedback
personalized
content
API (capture)
API (retrieve)
pre-computed
content
Σ
periodically batch
train model
periodically
run models

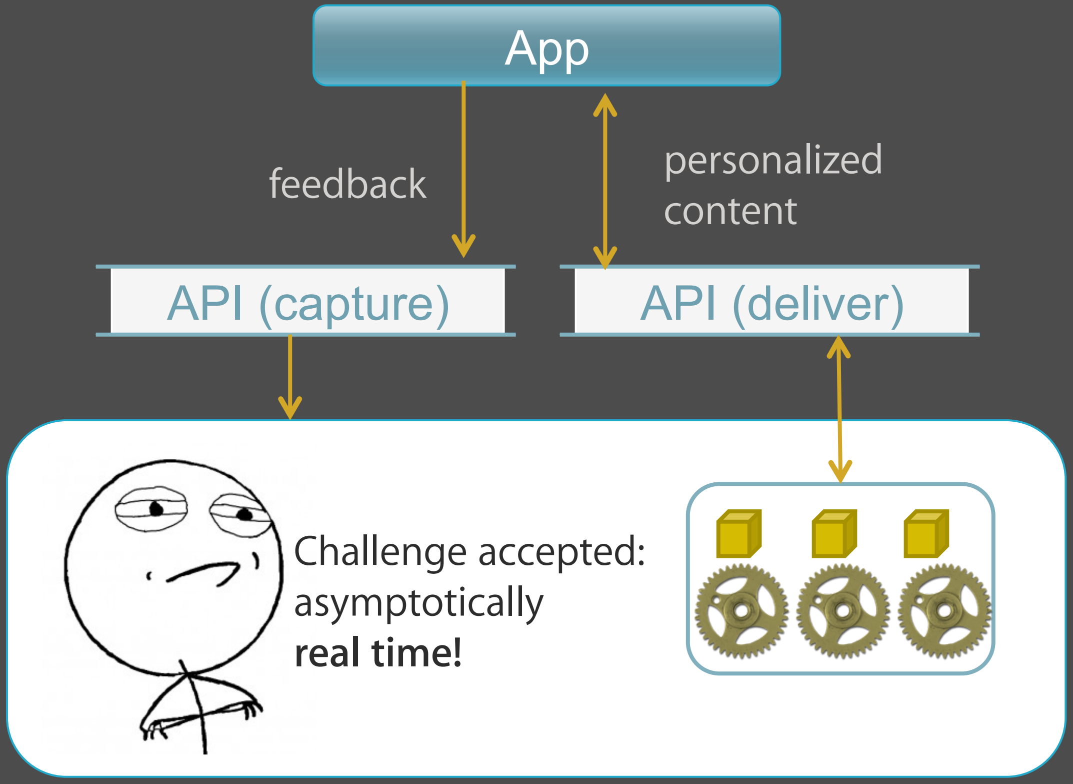
(B) On-the Fly


App
feedback
personalized content
API (capture)
API (compute)
compute on-the-fly
Σ
periodically batch train model

(C) Aggressive



(C) Aggressive



Maturity. Patterns and Assumptions.

Model Building

Model Deployment

Data Store

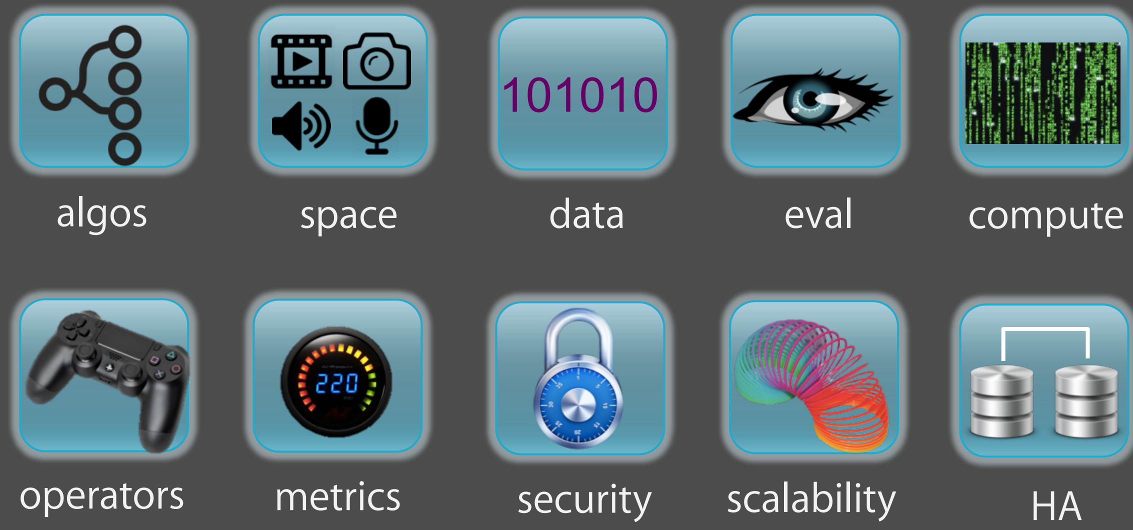
Content Delivery

Analytics

Data Capture

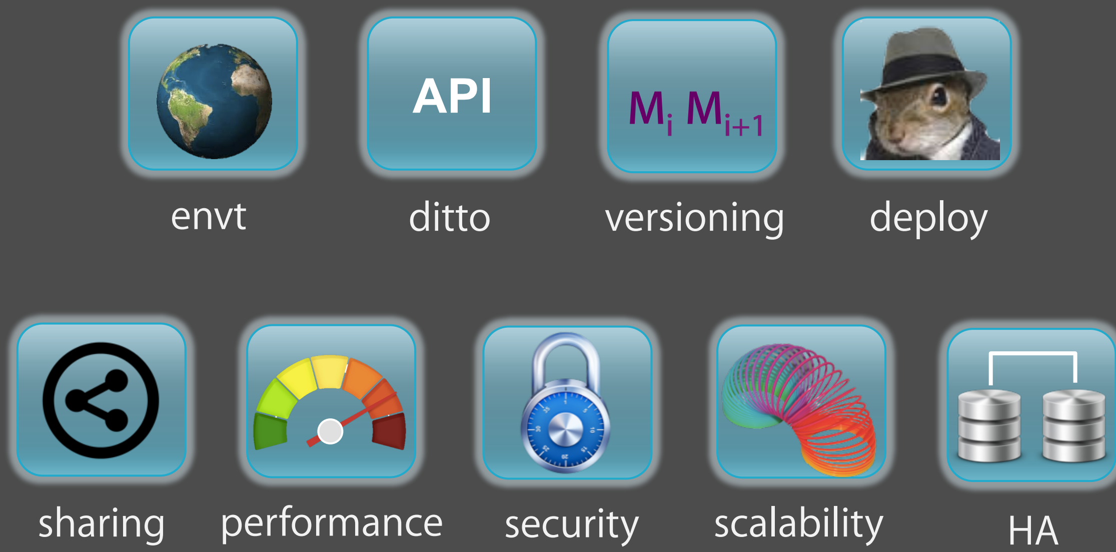
What do you *really* need?
Do you need it *now*?

Model Building. What do you *really* need?



Model Building. What do you *really* need?

Model Deployment. What do you *really* need?



Personalization Delivery. What do you *really* need?

Jabba

idea — Intuit Data Engineering & Analytics

Experiments | Priority | Users | Applications | Feedback | Welcome, mmahadevan1 | Sign out | Feedback

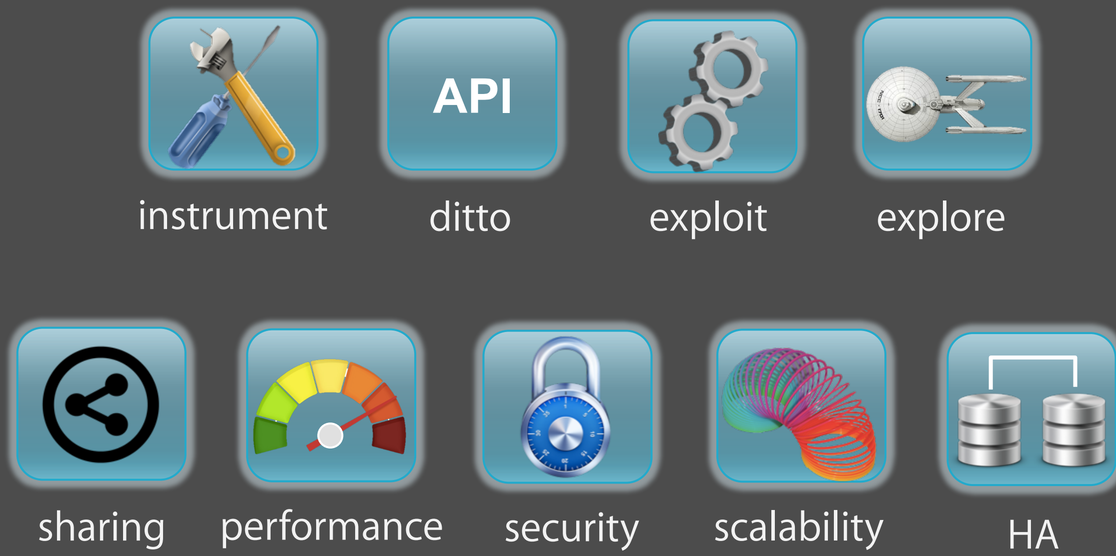
Create Experiment

Hide terminated | Search | +

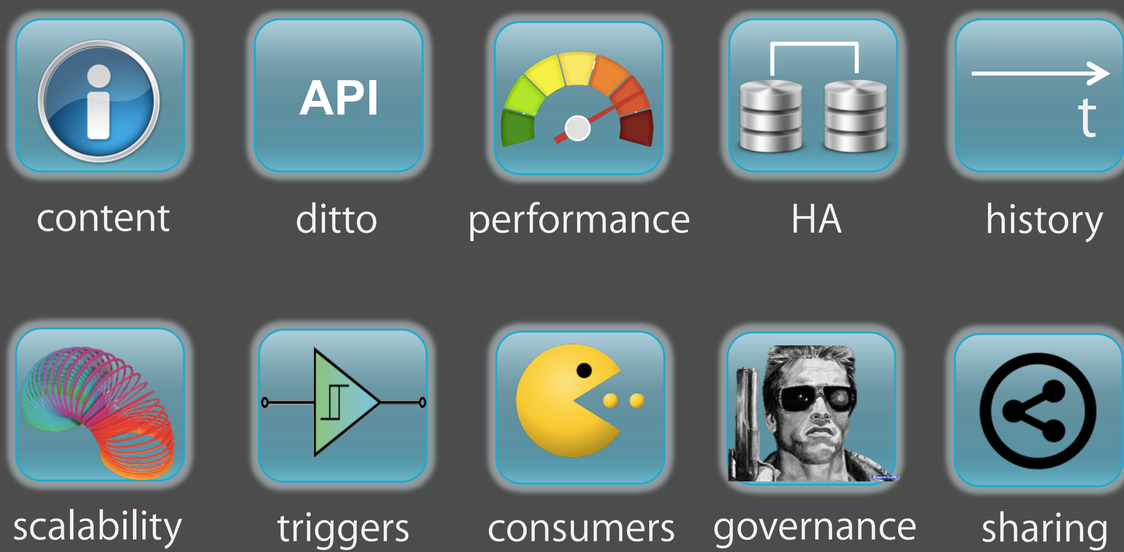
APP ▲	EXPERIMENT	SAMPLING %	START	END	MODIFIED	STATUS	ACTIONS
QBO	registers-us-qa	100	Jul 22, 2015	Dec 30, 2015	Jul 29, 2015		
QBO	lightning-bolt-ipd-201505-qa	100	Apr 27, 2015	Jul 31, 2015	Jun 02, 2015		
QBO	sangria-gulp4	100	Jun 08, 2014	Jul 08, 2014	Jun 08, 2014		
QBO	HomepagePlugin_1-qa	0.1	Jan 22, 2015	Jan 23, 2016	Jan 29, 2015		
QBO	HomepagePlugin_20150311-dev	75	Feb 28, 2015	Feb 28, 2016	Apr 06, 2015		
QBO	ProAssistPersonalPro_2015052...	0.01	Jun 15, 2015	Jan 01, 2016	Jun 16, 2015		
QBO	lightning-bolt-ipd-201505-e2e	100	Apr 27, 2015	Aug 31, 2015	Apr 27, 2015		
QBO	InvoiceIntuit-Show-Modal-dev	100	Aug 20, 2015	Aug 31, 2015	Aug 20, 2015		
QBO	ipd-zero-state-dtx-201505-e2e	100	May 01, 2015	Dec 30, 2015	Jul 29, 2015		
QBO	lightning-bolt-ipd-201405-prod	100	Apr 27, 2015	Apr 28, 2015	May 06, 2015		

511 - 520 of 647 | Previous | ... | 51 | 52 | 53 | 54 | 55 | ... | Next

Personalization Delivery. What do you *really* need?



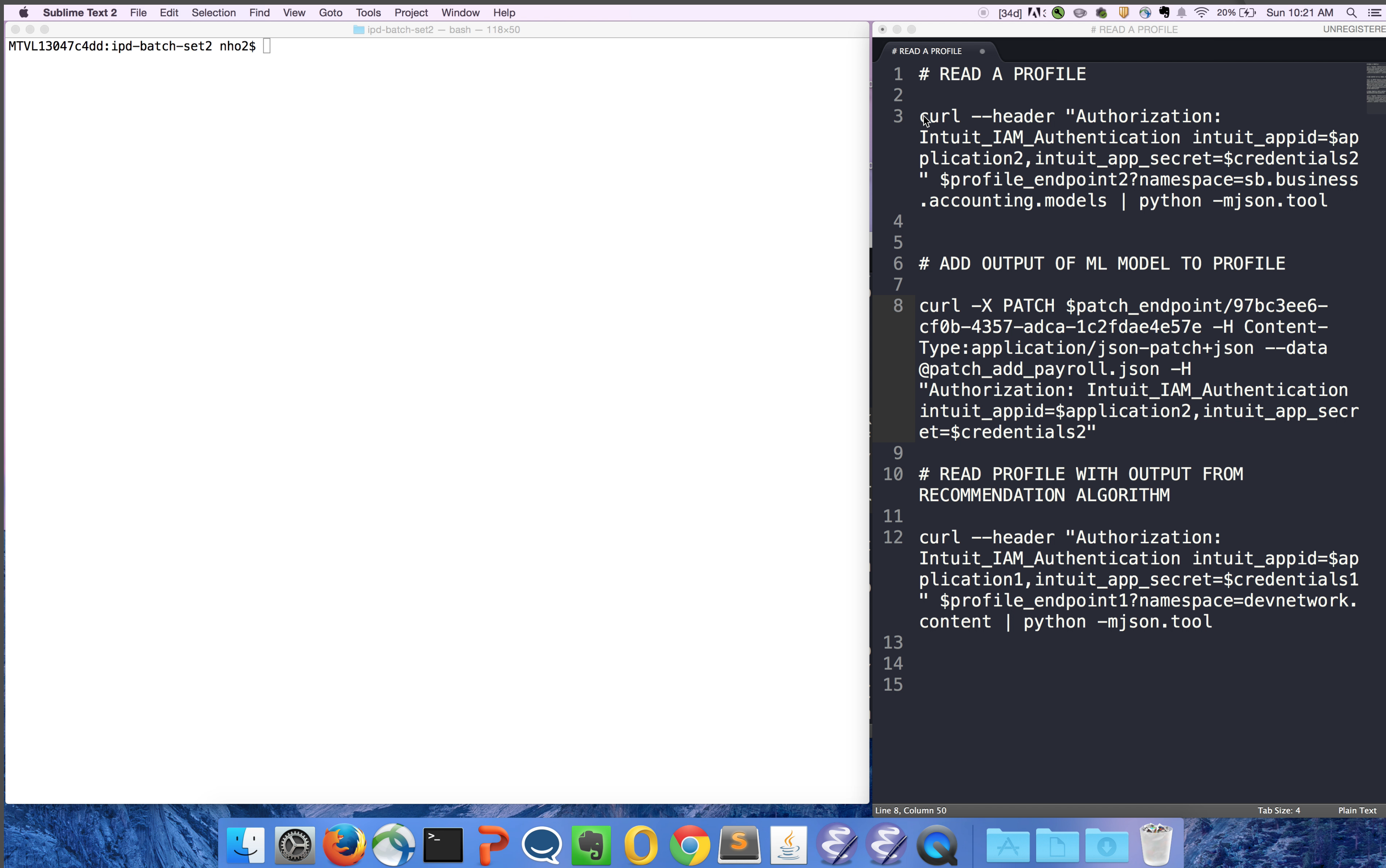
Data Store. What do you *really* need?



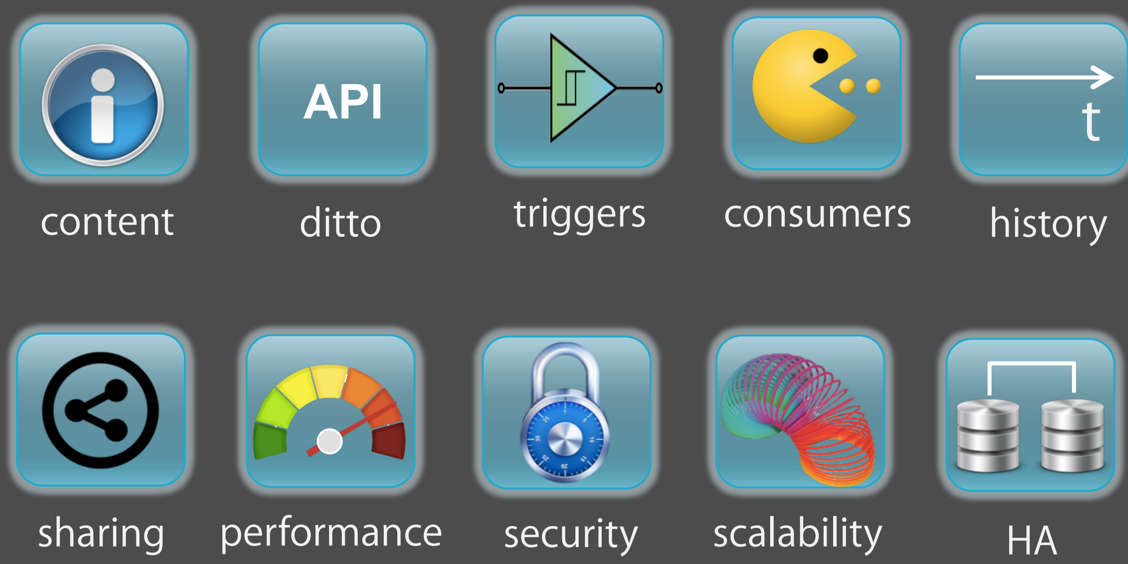
Data Store. To HA or not to HA.

now		later (blasphemy)	
	revenue driver		in-app
	critical user benefit		infrastructure cost
	known use cases		build & operate

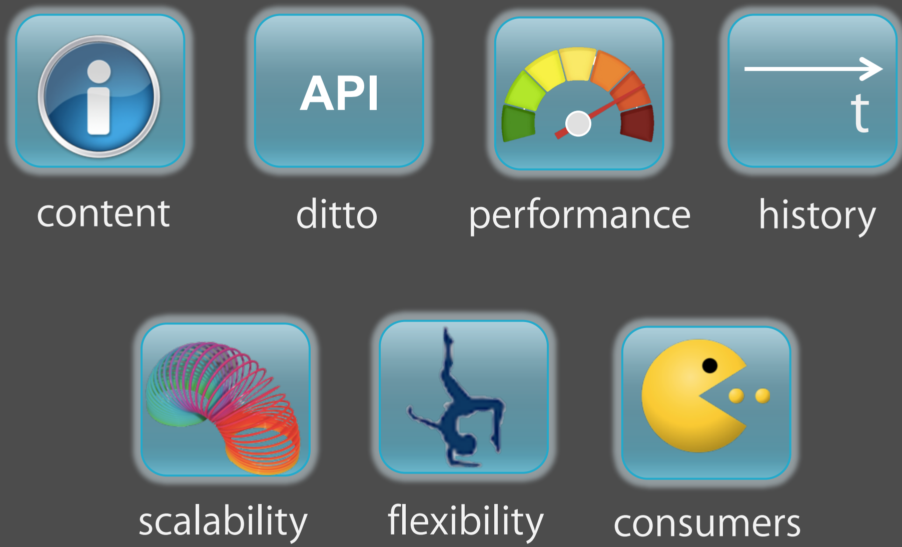
Data Store. APIs



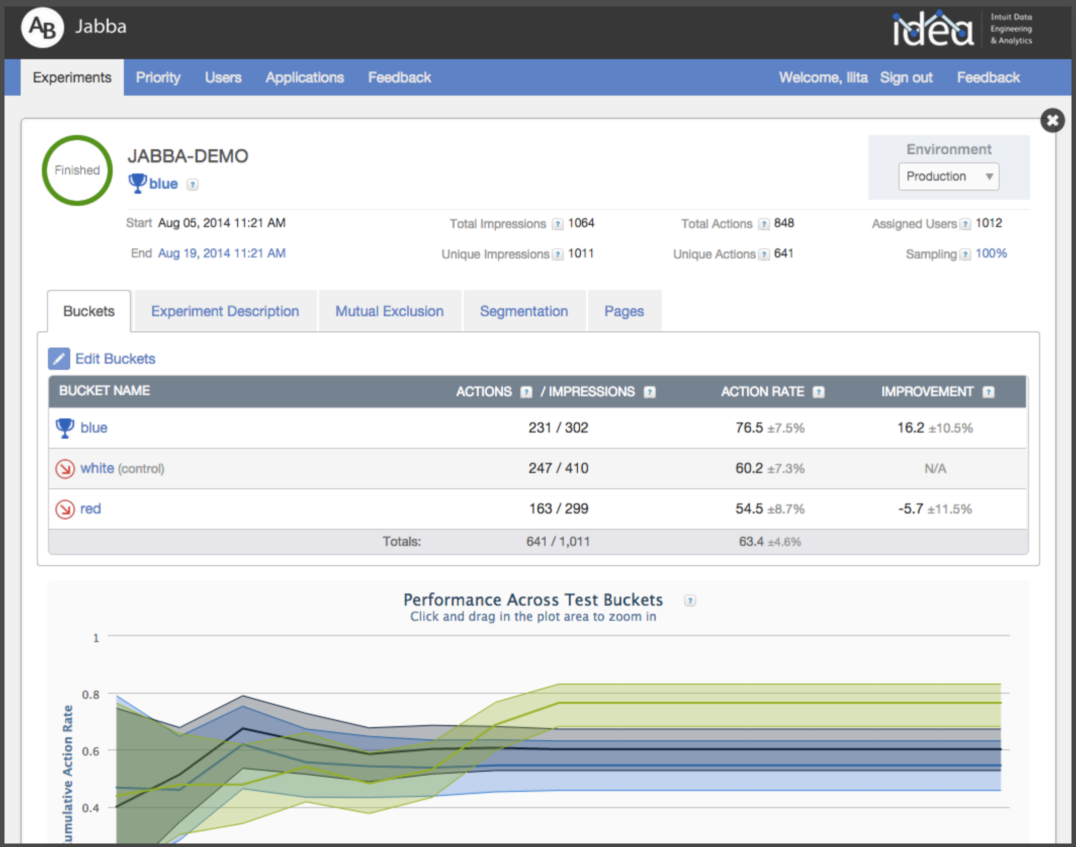
Data Capture. What do you *really* need?



Analytics. What do you *really* need?



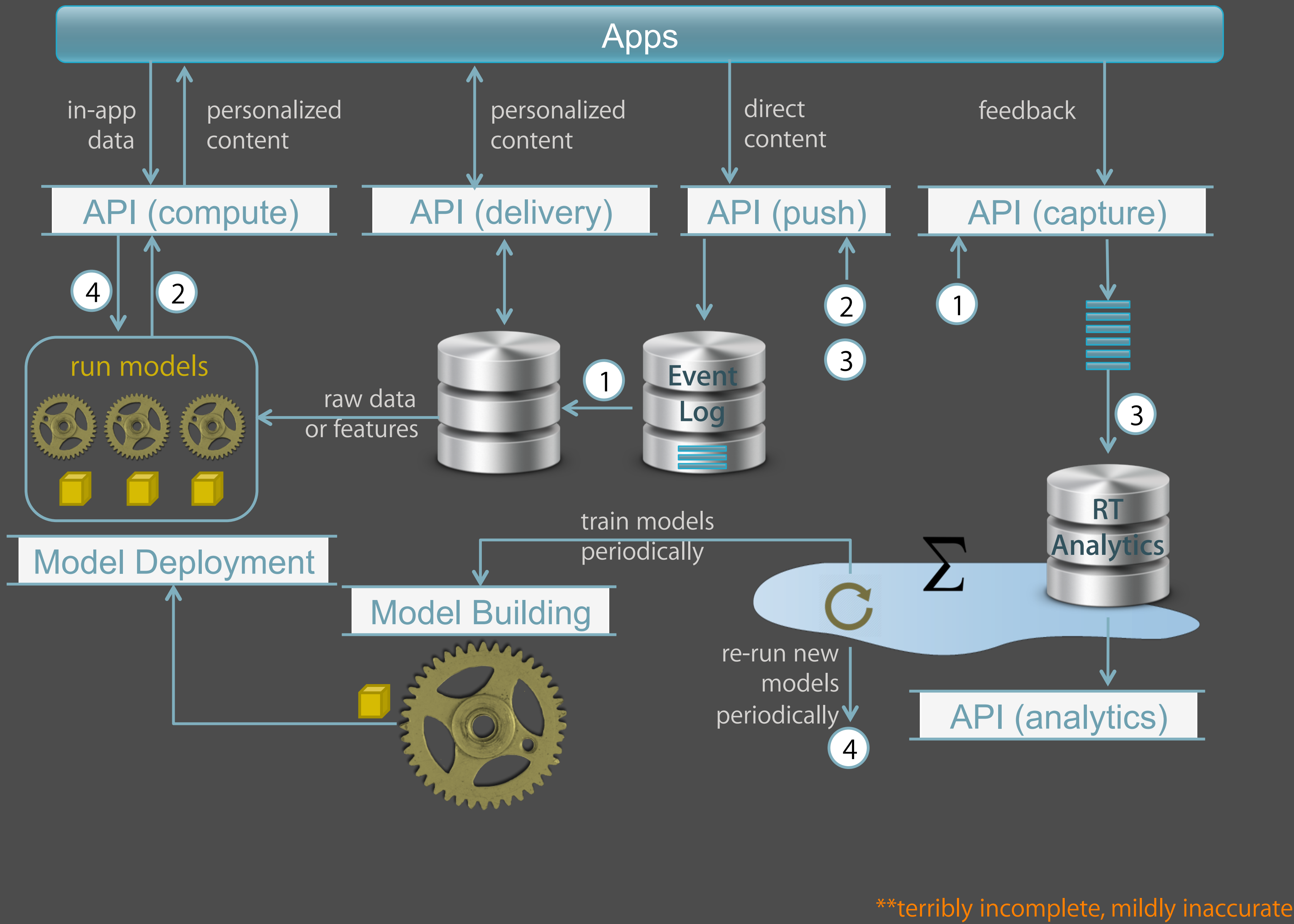
Analytics. Experimentation & Personalization

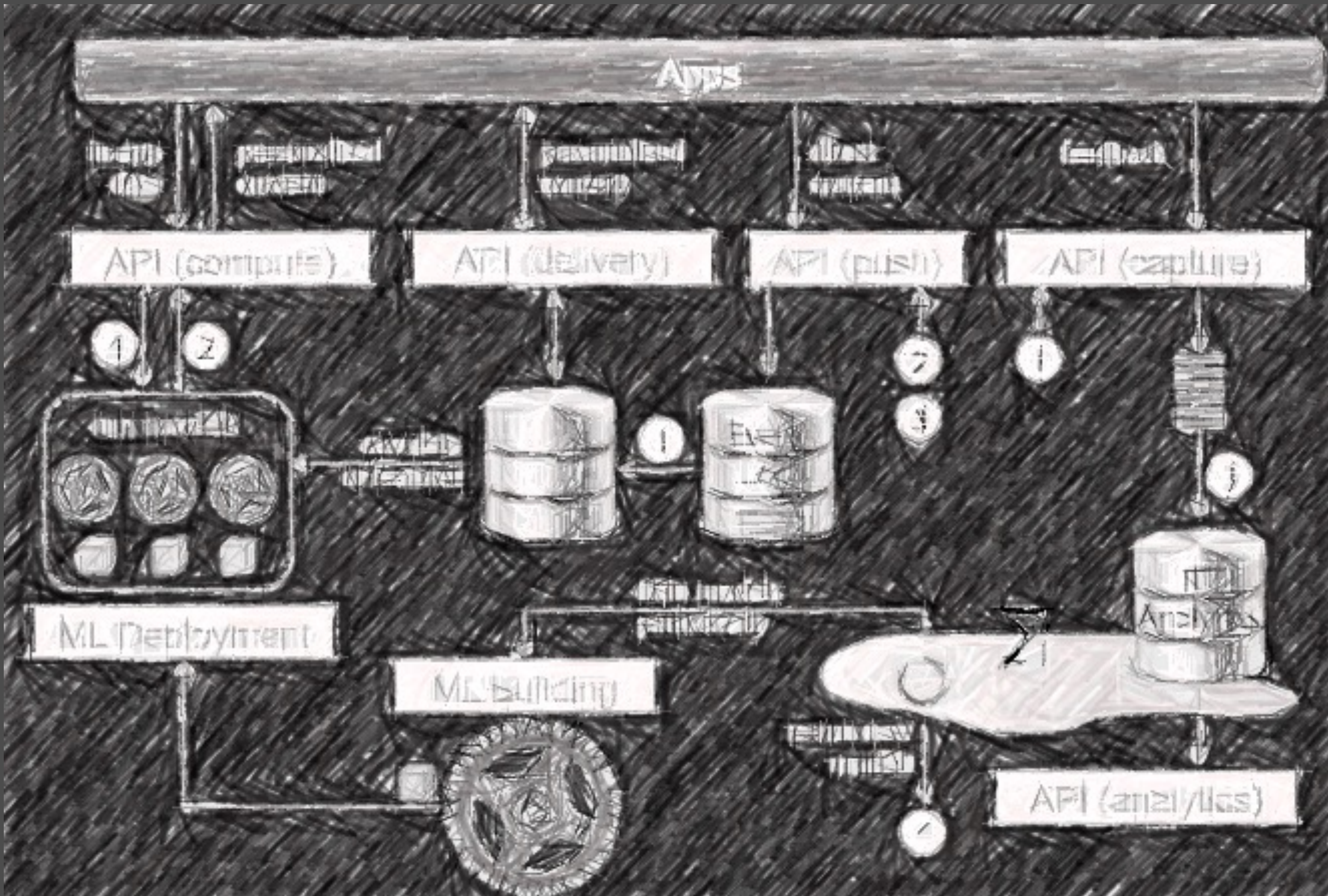


Data Lake. What do you *really* need?

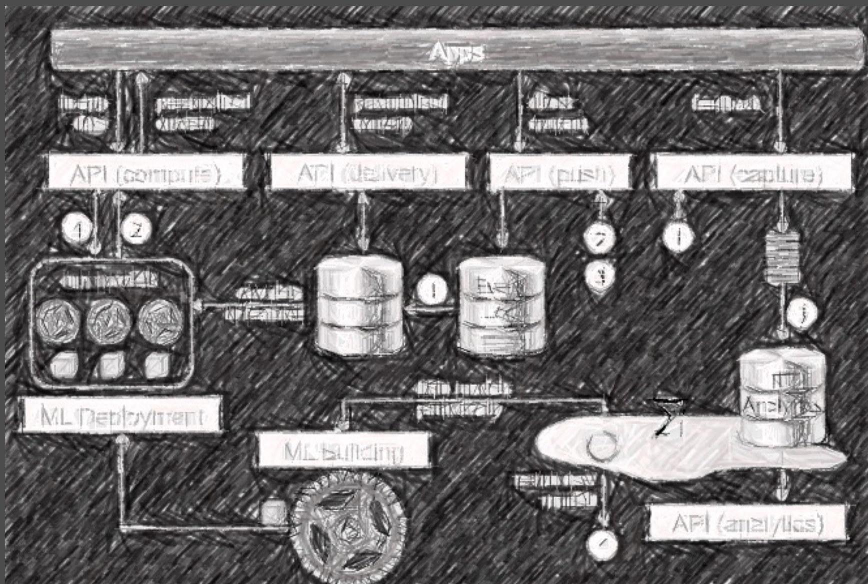
say 'big data lake'
one more time!

Evolving Architecture. Before you know it…


Apps
in-app data
personalized content
personalized content
direct content
feedback
API (compute)
API (delivery)
API (push)
API (capture)
4
2
2
1
3
1
3
run models
raw data or features
Event Log
RT Analytics
train models periodically
Model Deployment
Model Building
re-run new models periodically
4
API (analytics)
**terribly incomplete, mildly inaccurate



Not an Exact Blueprint



As you embark …

Know this

- non-trivial
- no one-size fits all

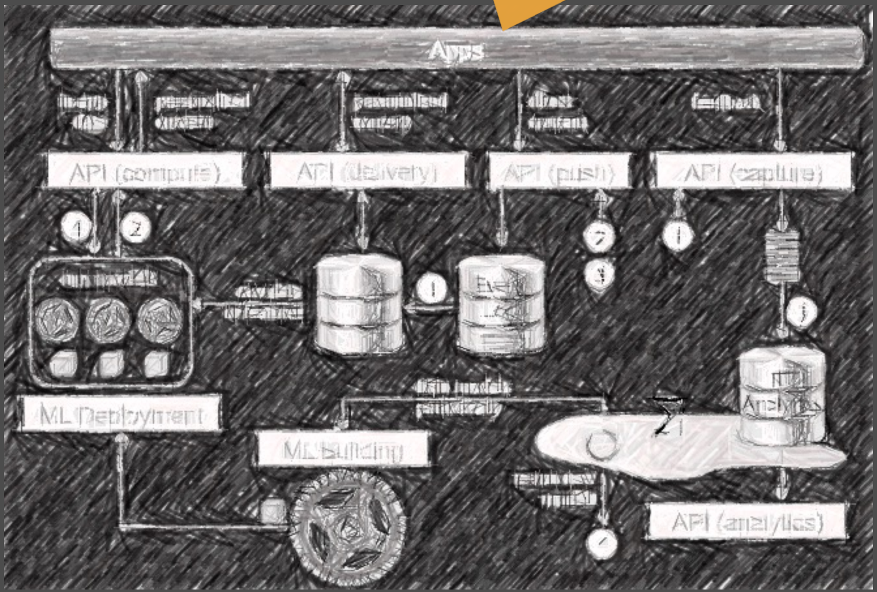
Upfront

- what do you really need?
- know thy target architecture

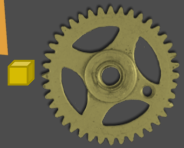
Do it!

- working system in weeks
- fast iterations – ship & test
- interfaaaaaaaces!

village



model



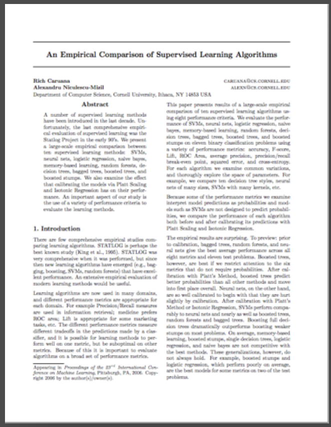
**not drawn to effort scale

Software architecture is the next frontier!

Fail fast still applies!

Personalize your personalization platform!

better algorithms



An Empirical Comparison of Supervised Learning Algorithms

Rich Caruana CARUANA@CS.CORNELL.EDU
Alexandru Niculescu-Mizil ALEXN@CS.CORNELL.EDU
Department of Computer Science, Cornell University, Ithaca, NY 14853 USA

Abstract

A number of supervised learning methods have been introduced in the last decade. Unfortunately, the last comprehensive empirical evaluation of supervised learning was the Statlog Project in the early 90's. We present a large-scale empirical comparison between ten supervised learning methods: SVMs, neural nets, logistic regression, naive bayes, memory-based learning, random forests, decision trees, bagged trees, boosted trees, and boosted stumps. We also examine the effect that calibrating the models via Platt Scaling and Isotonic Regression has on their performance. An important aspect of our study is the use of a variety of performance criteria to evaluate the learning methods.

1. Introduction

There are few comprehensive empirical studies comparing learning algorithms. STATLOG is perhaps the best known study (King et al., 1995). STATLOG was very comprehensive when it was performed, but since then new learning algorithms have emerged (e.g., bagging, boosting, SVMs, random forests) that have excellent performance. An extensive empirical evaluation of modern learning methods would be useful.

Learning algorithms are now used in many domains, and different performance metrics are appropriate for each domain. For example Precision/Recall measures are used in information retrieval; medicine prefers ROC area; Lift is appropriate for some marketing tasks, etc. The different performance metrics measure different tradeoffs in the predictions made by a classifier, and it is possible for learning methods to perform well on one metric, but be suboptimal on other metrics. Because of this it is important to evaluate algorithms on a broad set of performance metrics.

This paper presents results of a large-scale empirical comparison of ten supervised learning algorithms using eight performance criteria. We evaluate the performance of SVMs, neural nets, logistic regression, naive bayes, memory-based learning, random forests, decision trees, bagged trees, boosted trees, and boosted stumps on eleven binary classification problems using a variety of performance metrics: accuracy, F-score, Lift, ROC Area, average precision, precision/recall break-even point, squared error, and cross-entropy. For each algorithm we examine common variations, and thoroughly explore the space of parameters. For example, we compare ten decision tree styles, neural nets of many sizes, SVMs with many kernels, etc.

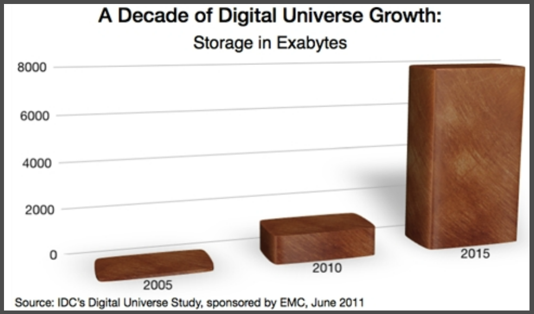
Because some of the performance metrics we examine interpret model predictions as probabilities and models such as SVMs are not designed to predict probabilities, we compare the performance of each algorithm both before and after calibrating its predictions with Platt Scaling and Isotonic Regression.

The empirical results are surprising. To preview: prior to calibration, bagged trees, random forests, and neural nets give the best average performance across all eight metrics and eleven test problems. Boosted trees, however, are best if we restrict attention to the six metrics that do not require probabilities. After calibration with Platt's Method, boosted trees predict better probabilities than all other methods and move into first place overall. Neural nets, on the other hand, are so well calibrated to begin with that they are hurt slightly by calibration. After calibration with Platt's Method or Isotonic Regression, SVMs perform comparably to neural nets and nearly as well as boosted trees, random forests and bagged trees. Boosting full decision trees dramatically outperforms boosting weaker stumps on most problems. On average, memory-based learning, boosted stumps, single decision trees, logistic regression, and naive bayes are not competitive with the best methods. These generalizations, however, do not always hold. For example, boosted stumps and logistic regression, which perform poorly on average, are the best models for some metrics on two of the test problems.

Appearing in *Proceedings of the 23rd International Conference on Machine Learning*, Pittsburgh, PA, 2006. Copyright 2006 by the author(s)/owner(s).

+1

more, better, smarter data



+1

well designed software architectures

+1

next frontier

intuit

A Brief Look at Anomaly Detection

Applications

- System health – servers, network
- Cyber-intrusion detection
- Enterprise anomaly detection
- Image processing
- Textual anomaly detection
- Sensor networks
- Fraud detection
- Medical anomaly detection
- Industrial damage detection
- …

Algorithms

A modified version of this technical report will appear in ACM Computing Surveys, September 2009.

Anomaly Detection : A Survey

VARUN CHANDOLA
University of Minnesota
ARINDAM BANERJEE
University of Minnesota
and
VIPIN KUMAR
University of Minnesota

Anomaly detection is an important problem that has been researched within diverse research areas and application domains. Many anomaly detection techniques have been specifically developed for certain application domains, while others are more generic. This survey tries to provide a structured and comprehensive overview of the research on anomaly detection. We have grouped existing techniques into different categories based on the underlying approach adopted by each technique. For each category we have identified key assumptions, which are used by the techniques to differentiate between normal and anomalous behavior. When applying a given technique to a particular domain, these assumptions can be used as guidelines to assess the effectiveness of the technique in that domain. For each category, we provide a basic anomaly detection technique, and then show how the different existing techniques in that category are variants of the basic technique. This template provides an easier and succinct understanding of the techniques belonging to each category. Further, for each category, we identify the advantages and disadvantages of the techniques in that category. We also provide a discussion on the computational complexity of the techniques since it is an important issue in real application domains. We hope that this survey will provide a better understanding of the different directions in which research has been done on this topic, and how techniques developed in one area can be applied in domains for which they were not intended to begin with.

Categories and Subject Descriptors: H.2.8 [**Database Management**]: Database Applications—*Data Mining*

General Terms: Algorithms

Additional Key Words and Phrases: Anomaly Detection, Outlier Detection

1. INTRODUCTION

Anomaly detection refers to the problem of finding patterns in data that do not conform to expected behavior. These non-conforming patterns are often referred to as anomalies, outliers, discordant observations, exceptions, aberrations, surprises, peculiarities or contaminants in different application domains. Of these, anomalies and outliers are two terms used most commonly in the context of anomaly detection; sometimes interchangeably. Anomaly detection finds extensive use in a wide variety of applications such as fraud detection for credit cards, insurance or health care, intrusion detection for cyber-security, fault detection in safety critical systems, and military surveillance for enemy activities.

The importance of anomaly detection is due to the fact that anomalies in data translate to significant (and often critical) actionable information in a wide variety of application domains. For example, an anomalous traffic pattern in a computer

To Appear in ACM Computing Surveys, 09 2009, Pages 1–72.

- Supervised
- Unsupervised
- Generic statistical
- Information theory
- …

"What algorithms are you going to use?"

Data

Low data volume

- Invest in data acquisition
- Invest in high coverage

High data volume

- Invest in defining signal
- Invest in labeling, tools, and crowdsourcing

Architectures Again

Data Collectors
Clickstream, User Input …
Real time, DBs …

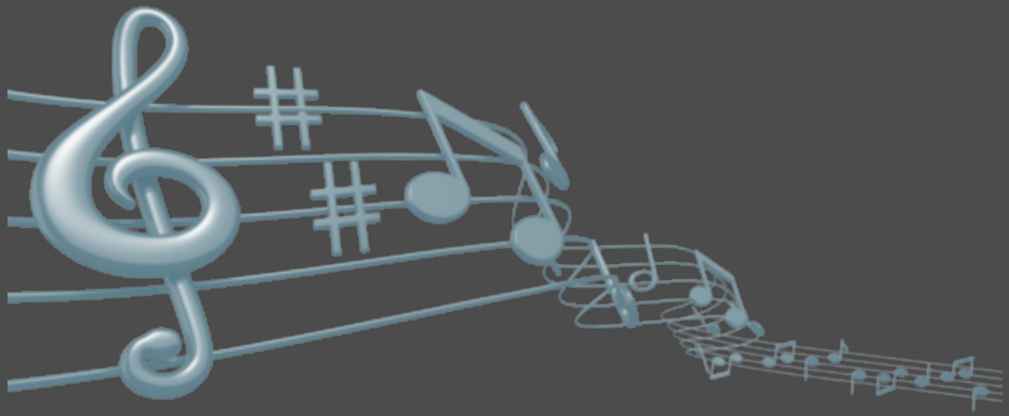
Labeling
Crowdsourcing
Active learning

Processors (M&A)
broad: time bounded
deep: open ended

**check assumptions

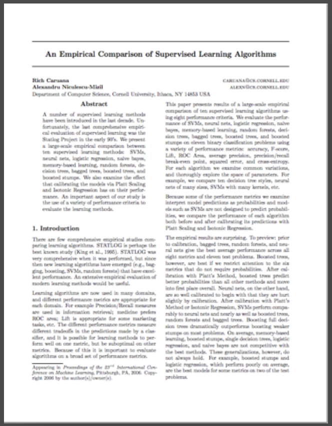
Advertising

Music Streaming



Medical Informatics

better
algorithms



An Empirical Comparison of Supervised Learning Algorithms

Rich Caruana CARUANA@CS.CORNELL.EDU
Alexandru Niculescu-Mizil ALEXN@CS.CORNELL.EDU
Department of Computer Science, Cornell University, Ithaca, NY 14853 USA

Abstract

A number of supervised learning methods have been introduced in the last decade. Unfortunately, the last comprehensive empirical evaluation of supervised learning was the Statlog Project in the early 90's. We present a large-scale empirical comparison between ten supervised learning methods: SVMs, neural nets, logistic regression, naive bayes, memory-based learning, random forests, decision trees, bagged trees, boosted trees, and boosted stumps. We also examine the effect that calibrating the models via Platt Scaling and Isotonic Regression has on their performance. An important aspect of our study is the use of a variety of performance criteria to evaluate the learning methods.

1. Introduction

There are few comprehensive empirical studies comparing learning algorithms. STATLOG is perhaps the best known study (King et al., 1995). STATLOG was very comprehensive when it was performed, but since then new learning algorithms have emerged (e.g., bagging, boosting, SVMs, random forests) that have excellent performance. An extensive empirical evaluation of modern learning methods would be useful.

Learning algorithms are now used in many domains, and different performance metrics are appropriate for each domain. For example Precision/Recall measures are used in information retrieval; medicine prefers ROC area; Lift is appropriate for some marketing tasks, etc. The different performance metrics measure different tradeoffs in the predictions made by a classifier, and it is possible for learning methods to perform well on one metric, but be suboptimal on other metrics. Because of this it is important to evaluate algorithms on a broad set of performance metrics.

Appearing in *Proceedings of the 23rd International Conference on Machine Learning*, Pittsburgh, PA, 2006.

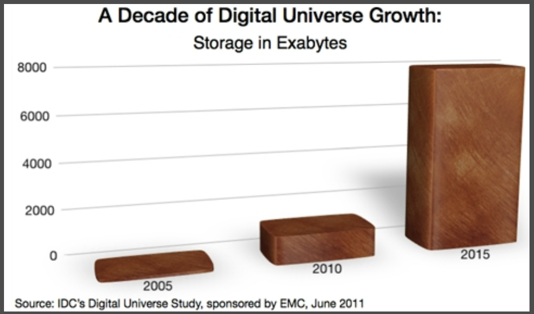
This paper presents results of a large-scale empirical comparison of ten supervised learning algorithms using eight performance criteria. We evaluate the performance of SVMs, neural nets, logistic regression, naive bayes, memory-based learning, random forests, decision trees, bagged trees, boosted trees, and boosted stumps on eleven binary classification problems using a variety of performance metrics: accuracy, F-score, Lift, ROC Area, average precision, precision/recall break-even point, squared error, and cross-entropy. For each algorithm we examine common variations, and thoroughly explore the space of parameters. For example, we compare ten decision tree styles, neural nets of many sizes, SVMs with many kernels, etc.

Because some of the performance metrics we examine interpret model predictions as probabilities and models such as SVMs are not designed to predict probabilities, we compare the performance of each algorithm both before and after calibrating its predictions with Platt Scaling and Isotonic Regression.

The empirical results are surprising. To preview: prior to calibration, bagged trees, random forests, and neural nets give the best average performance across all eight metrics and eleven test problems. Boosted trees, however, are best if we restrict attention to the six metrics that do not require probabilities. After calibration with Platt's Method, boosted trees predict better probabilities than all other methods and move into first place overall. Neural nets, on the other hand, are so well calibrated to begin with that they are hurt slightly by calibration. After calibration with Platt's Method or Isotonic Regression, SVMs perform comparably to neural nets and nearly as well as boosted trees, random forests and bagged trees. Boosting full decision trees dramatically outperforms boosting weaker stumps on most problems. On average, memory-based learning, boosted stumps, single decision trees, logistic regression, and naive bayes are not competitive with the best methods. These generalizations, however, do not always hold. For example, boosted stumps and logistic regression, which perform poorly on average, are the best models for some metrics on two of the test problems.

+1

more, better, smarter
data



+1

well designed
software architectures

+1

next frontier

Thank you!

Lucian Lita

@datariver

[always hiring]

Thank you!

Lucian Lita
@datariver

[always hiring]

Extra Content

Security. What do you *really* need?

Jabba

Experiments | Priority | Users | Applications | Feedback | Welcome, Ilita | Sign out | Feedback

Finished

JABBA-DEMO

blue

Environment: Production

Start Aug 05, 2014 11:21 AM	Total Impressions 1064	Total Actions 848	Assigned Users 1012
End Aug 19, 2014 11:21 AM	Unique Impressions 1011	Unique Actions 641	Sampling 100%

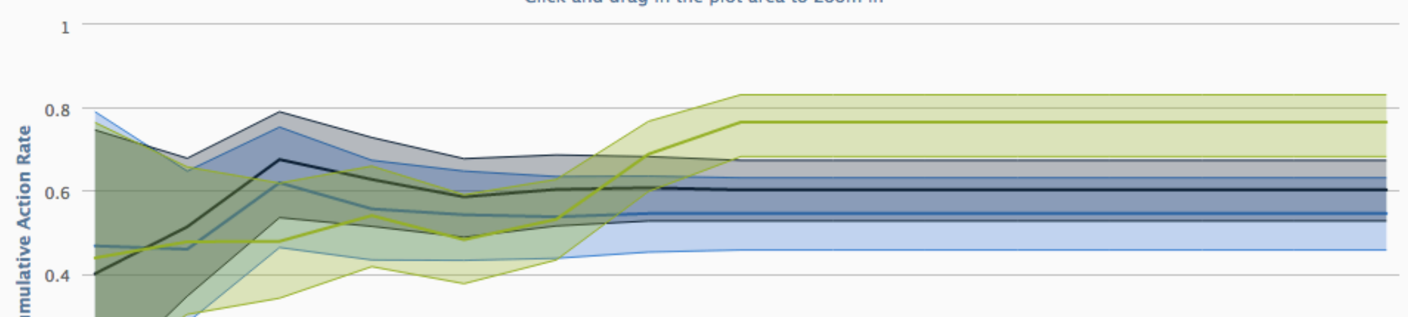
Buckets | Experiment Description | Mutual Exclusion | Segmentation | Pages

Edit Buckets

BUCKET NAME	ACTIONS / IMPRESSIONS	ACTION RATE	IMPROVEMENT
blue	231 / 302	76.5 ±7.5%	16.2 ±10.5%
white (control)	247 / 410	60.2 ±7.3%	N/A
red	163 / 299	54.5 ±8.7%	-5.7 ±11.5%
Totals:	641 / 1,011	63.4 ±4.6%	

Performance Across Test Buckets

Click and drag in the plot area to zoom in



App. Who does the App talk to?

(a)

App

personalized
content

API (retrieve)

-- apply op logic
-- retrieve pre-computed
content

(b)

App

dynamic
data

personalized
content

API (compute)

-- retrieve static data
-- apply op logic
-- compute features
-- run model
-- log actions