Productionizing machine-learning services: Lessons from Google

{salim,villavieja}@google.com

SREcon Asia 2018

We are not machine learning hackers/ninjas
We are not machine learning scientists

We are **experienced SREs** and we have collected production insights through a **large number of interviews (~40)** from teams using ML in production at Google over the last 15 years.

Find the mistake about ML

ML is easy

ML is new

ML is a black box, no need to know more

Train “one and done”

You rarely rollback

ML monitoring is like any other monitoring

More data is better

Learn all the patterns

Transparent to user

Compatibility is a no-op

What is ML good for?

What is ML good for?

Everything!

Image source:
https://pixabay.com/en/businessman-boxes-transport-2108029/
CC0 license.

What is ML good for?

Everything!

Except when ...

- No fallback plan
- Not enough labeled data
- Requires microsecond latency

Image source: https://pixabay.com/en/businessman-boxes-transport-2108029/
CC0 license.

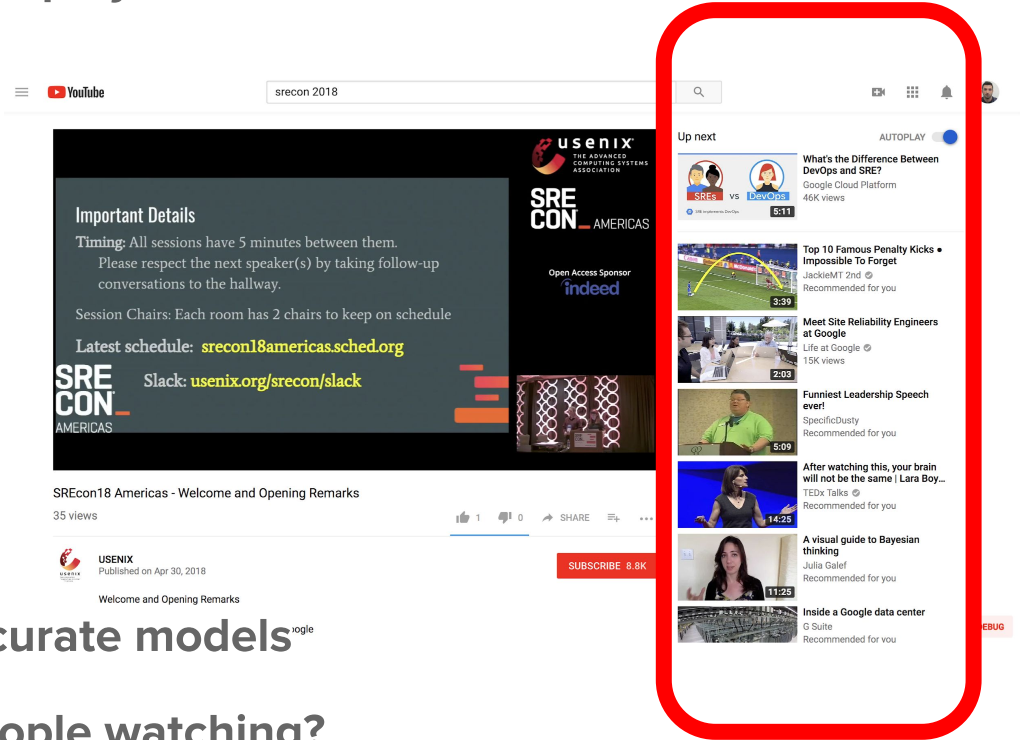
Some Google use cases of ML in production

Ads	Predict user clicks.
Prefetching	Predict next memory or next file access in large systems.
Resources/Sched	Predict RAM/CPU usage of jobs. Compaction in bigtable/databases.
Speech/Translate	Detect language, detect speaker, improve translation.
Fraud	Check credit cards and transactions.
Gmail	Suggest smart responses to all your emails.
Perception	Image and video understanding (Google Photos, YouTube and others)

One Very Important ML model At Google

Youtube ML for video recommendations

- **Continuously training & Fast deployment**
- **Keep high accuracy**
- **World Wide input data**
- **Revenue facing**
- **More video time +**
  More ad clicks
- **Special events one day**
- **User can easily detect not accurate models**
- **What's the fallback? Other people watching?**



But it's not that easy in production

GUARANTEE FRESHNESS

MULTIPLE DEVICES

MONITOR VIEW TIME/...

FILTERING SPAM/BAD VIDEOS

DEPLOYING EVERY *N* HOURS / DAYS / WEEKS

CONTINUOUSLY TRAINING MODELS

Our goals

Based on Google's

ML production teams:

ML best practices

OK Google:

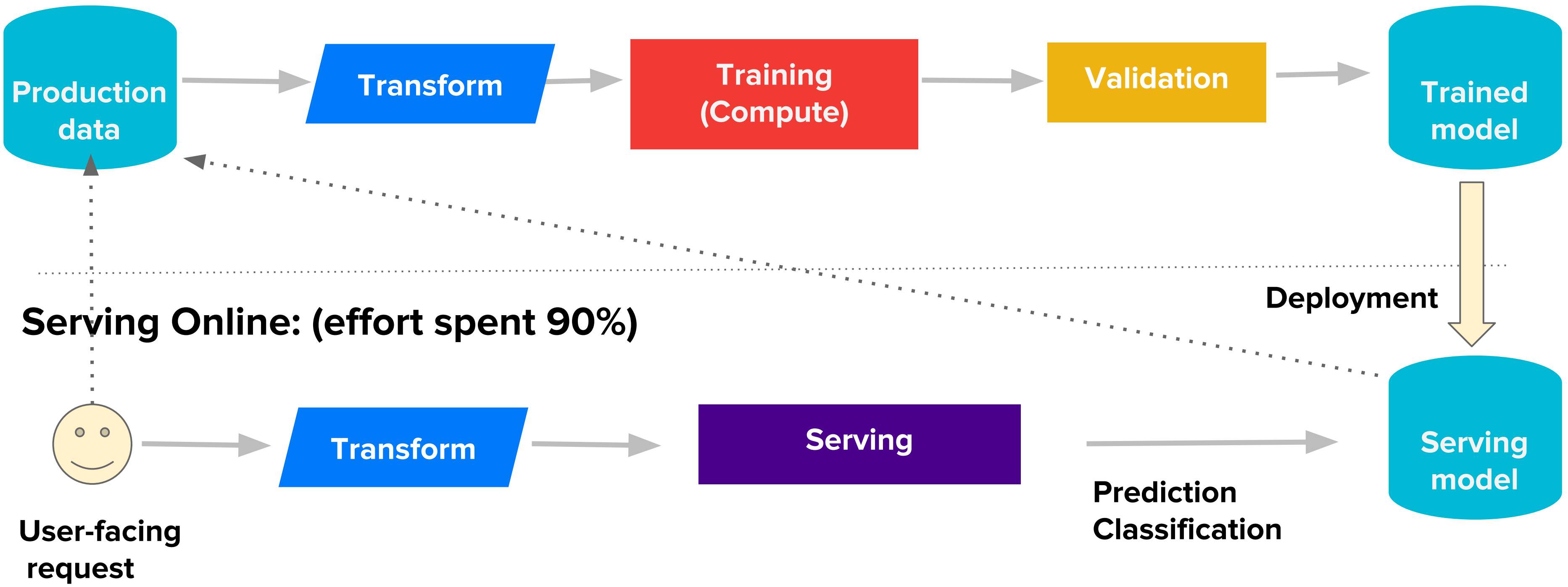
OK Google:
What's ML like in prod?

What's ML like in prod?

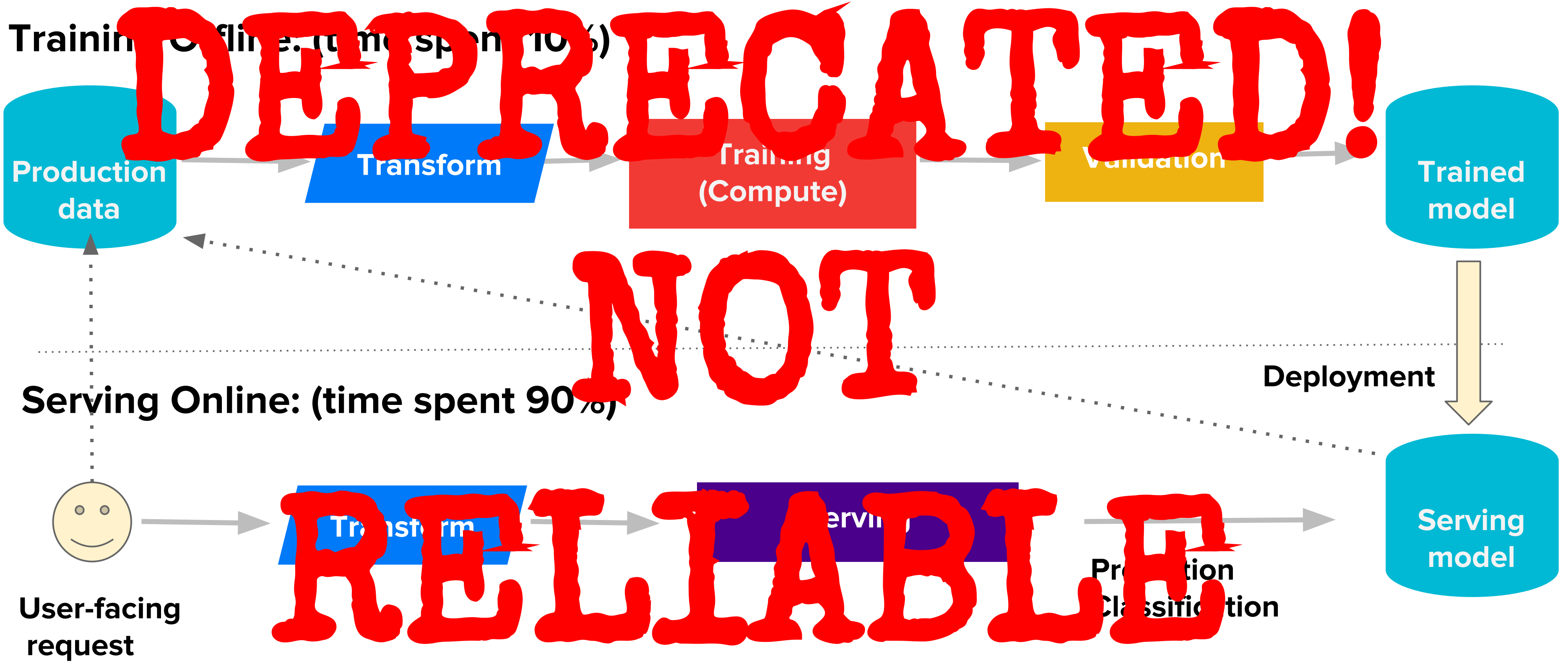
'It's just another data pipeline'

Theoretical Machine Learning Pipeline

Training Offline: (effort spent 10%)



Theoretical Machine Learning Pipeline



DESCRIBE BEST PRACTICES: Why are they important

Part 1 : TRAINING & DATA QUALITY	RELIABLE
Part 2 : HARDWARE RESOURCES (GPU/TPU)	FAST
Part 3 : QUALIFICATION	PROD READY
Part 4 : BACKWARDS COMPATIBILITY/CONF.MANAGEMENT	EASY
Part 5 : PRIVACY AND ETHICS	MUST

(re) Training

(not prototyping)

Training

Integral part of the release process

Not coding, debugging, testing
Input data coming to the training pipeline can't be stopped

Production changes fast:

Model loss increases with time at a constant rate.

Training

Production changes fast:

<u>Model loss</u> increases with time at a constant rate.

Training

Production changes fast:

<u>Model loss</u> increases with time at a constant rate.

Training

Production changes fast:

<u>Model loss</u> increases with time at a constant rate.

Training

Production changes fast:

<u>Model loss</u> increases with time at a constant rate.

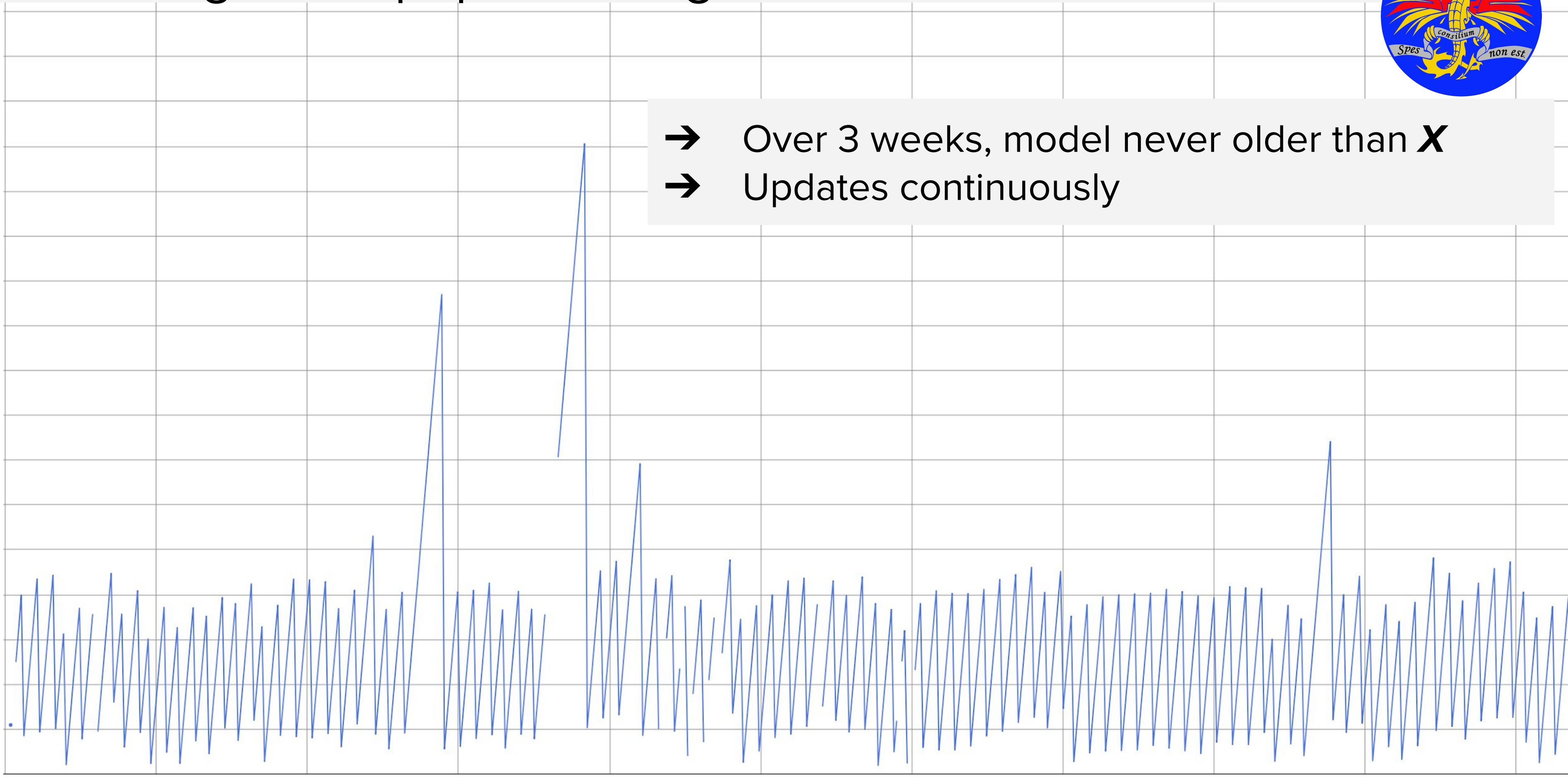
Training

Production changes fast:

<u>Model loss</u> increases with time at a constant rate.

Model Age for a popular Google Service

- ➔ Over 3 weeks, model never older than ***X***
- ➔ Updates continuously



Model Age for a popular Google Service

- Over 3 weeks, model never older than ***X***
- Updates continuously
-

Non-stop training

Models might need to evolve fast

Training: Filtering is key

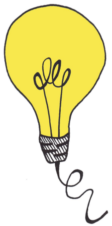
- **Good:** Train your model with ***all*** data, from oldest to newest
- **Bad:** We can't ***ALWAYS*** train on all production data. (Youtube 1.2 TB ML model)
- Production data has ***tons of duplicate information*** and needs to be filtered.
- **Filtering**: collapse duplicate values, to construct the model efficiently.
- **Data Imputation:** replacing missing data with substituted values

```
1, carlos, male,41, Spanish, 6.2, SRE, NULL, 80%
2, salim, male, 44, American, 5.8, SRE, +4, 90%
3, maria, female, 0, Norway, 6.0, SWE, +25, 60%
4, fep, agender, Spanish, 6.0, SWE, +5, 75%
5, maria, female, 0, Norway, 6.0, SWE, +25, 60%
```

Training: Filtering is key

- **Good:** Train your model with ***all*** data, from oldest to newest
- **Bad:** We can't ***ALWAYS*** train on all production data. (Youtube 1.2 TB ML model)
- Production data has ***tons of duplicate information*** and needs to be filtered.
- **Filtering**: collapse duplicate values, to construct the model efficiently.
- **Data Imputation:** replacing missing data with substituted values

Filter bad data, add data imputation on all fields



```
1, carlos, male,41, Spanish, 6.2, SRE, NULL, 80%
2, salim, male, 44, American, 5.8, SRE, +4, 90%
3, maria, female, 0, Norway, 6.0, SWE, +25, 60%
4, fep, agender, Spanish, 6.0, SWE, +5, 75%
5, maria, female, 0, Norway, 6.0, SWE, +25, 60%
```

Training: Data size

- Validation data is not the same as trained data
  - Trust that your high-accuracy model is correct with data not used during training.
    - 80-20/70-30 might vary depending on the model
    - Randomly selected set from the trained data
- Do not confuse with *qualification* (to be seen later)

No "one size fits all"

Image source: https://vimeo.com/122534562
License: https://creativecommons.org/licenses/by/3.0/

Training *at Scale*

- Very large data sets.
- How many models are continuously training (batch) ?
  - Different regions? Different time zones?
  - Available compute resources might be an issue.
- Snapshot your model:
  - Warm start on training
  - Avoid losing time if scheduled out

Image source: https://vimeo.com/122534562
License: https://creativecommons.org/licenses/by/3.0/

Summary: Data Quality on Training

FEATURES

All details about data that can be represented as a number

Summary: Data Quality on Training

CORRECT

Data imputation and data validation so that your models never receive unexpected inputs.

SNAPSHOTS	Train over previous models (resuming and rollback)
BIAS	Monitor amount of data from different sources. Features skews (train features diff from inference features)
ANOMALIES	Can't train with all SuperBowl day/New Years

COMPLETE

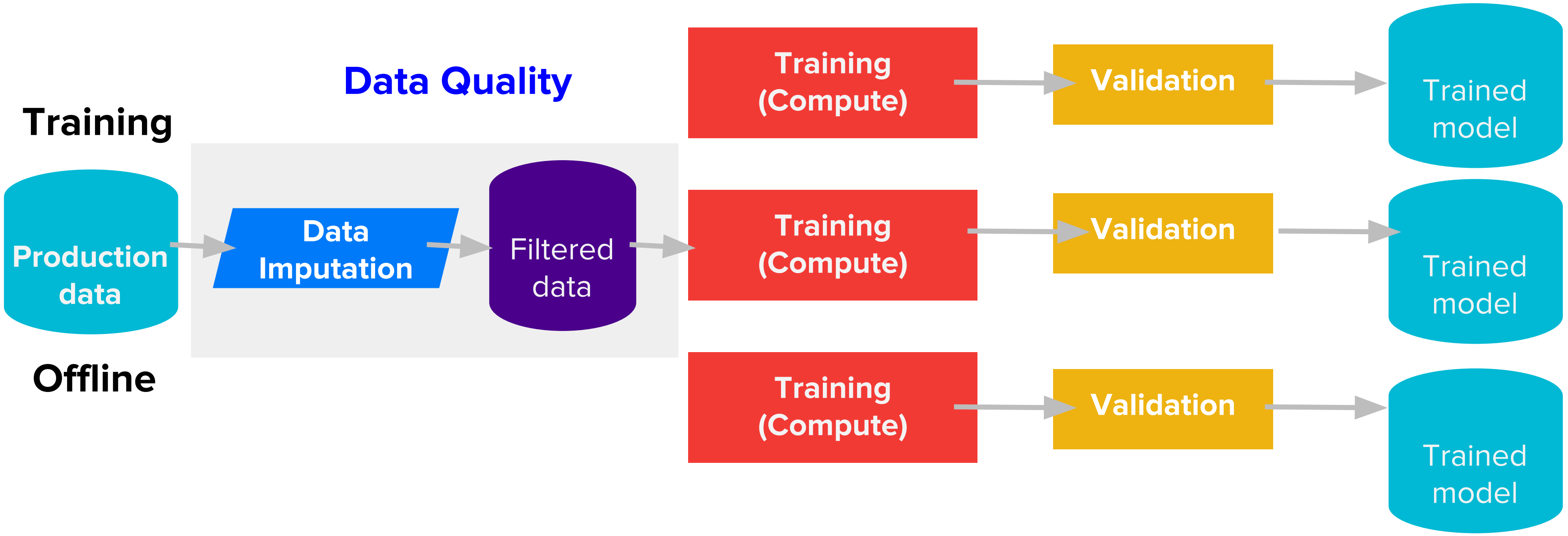
Missing inputs previously used

DATA RATIOS	Continent X pipeline stopped and the youtube recommendation models stops taking into account those videos.
AUTOMATION	Be ready to add fields on old data. Be ready to fix your data (spam data in trained models)

Before Machine Learning Pipeline

Training (time spent 10%)

After Machine Learning Pipeline



Resources

Training a large-scale machine translation model

24 hours on 32 GPUs

6 hours on a *fraction* of a TPU Pod

*slide source: [Cloud Discover: ML Workshop Presentation]

Why hardware resources are important

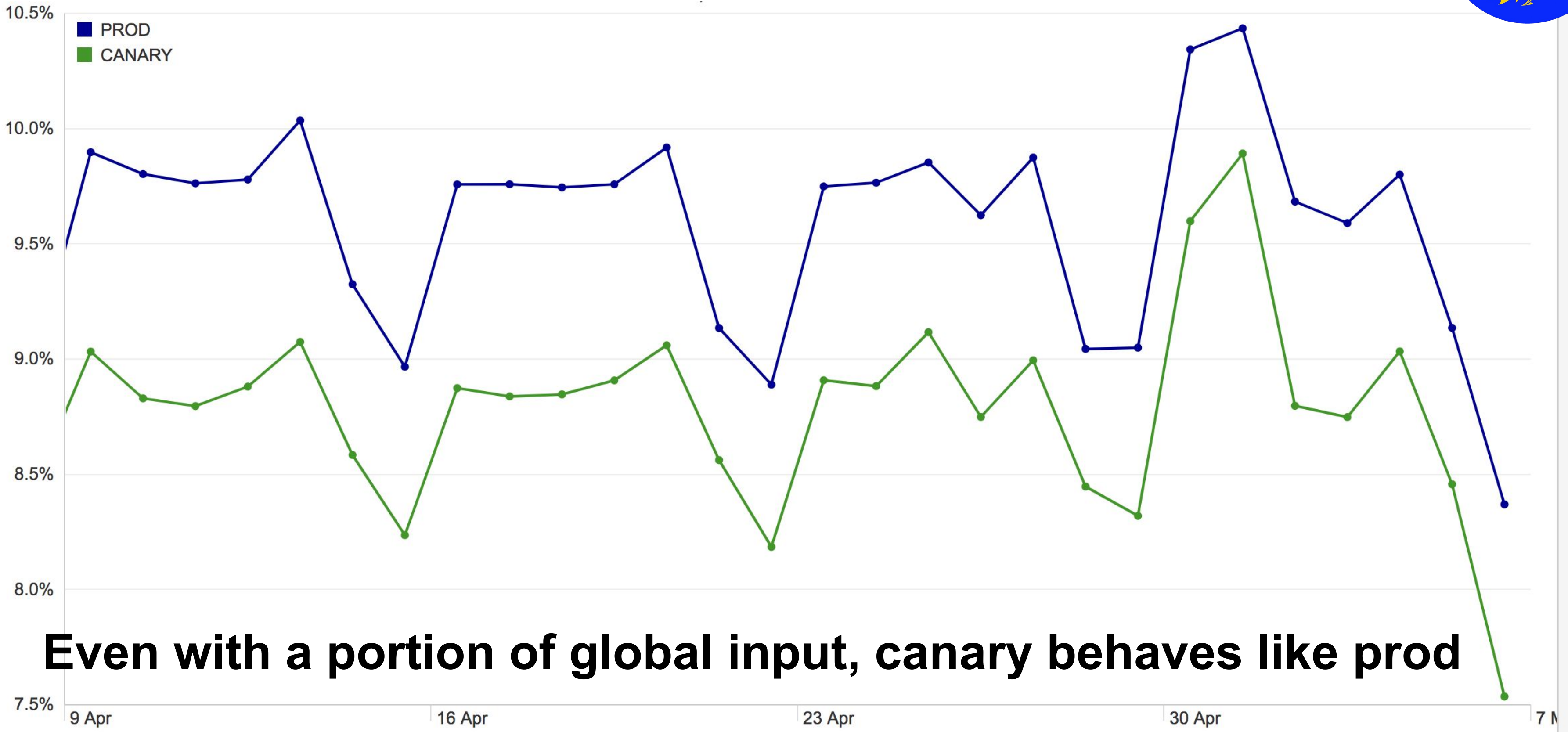
- ❑ Two different & disjoint environments
  - ❑ Training
  - ❑ Serving/Inference
- ❑ Cost of Training resources grows **at a higher rate** than Production resources

Qualification

Model Qualification

- ❑ Models are qualified with a separate input data
  - ❑ How is this data chosen? (previous or same prod day)
- ❑ **Models are tested with the same production binary.**
- ❑ Or we have an A/B testing scenario
  - ❑ Same production code/release
  - ❑ Dynamically decide % predictions to each model

Canary is a must



Even with a portion of global input, canary behaves like prod

Model Qualification

The model is signed post qualification.

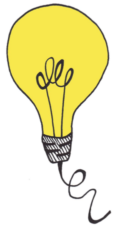
- ❑ Some providers allow to register models for versioning
- ❑ Signature specifies type of model, input/output data

Only allow *signed* models in production.

Backwards Compatibility

Backwards Compatibility

- Input data changes:
  - New fields, new values, null/empty values not contemplated.
- API changes:
  - Tensorflow API changes frequently, Incompatible model
  - "The model was completely valid and healthy as configured, it was simply not configured for the type of traffic it would receive"



- Fallback mechanisms, when rollback not an option:
  - Most teams do not have a non-ML fallback mechanisms
  - What happens if you run out of quota/capacity.
- **Old models need to be deprecated, they might not be reusable**
  - New inputs (labels) deployed
  - Signing the models helps on this. However, we're not able to ask the model compatibility?

Which config is running in prod?

- ❑ Do code and models go together? Are they deployed in the same package?
- ❑ How do we verify model and code compatibility?
  - ❑ *Always* push through canary
- ❑ What API version is this model for?

Rollbacks and Cloning

You can't add a new feature to an old model
(without re-training)

This limits backwards compatibility.

- ❑ Run it in canary
- ❑ Rollbacks must be easy
- ❑ Does a rollback involve human judgement?

Rollbacks and Cloning

You can't add a new feature to an old model
(without re-training)

This limits backwards compatibility.

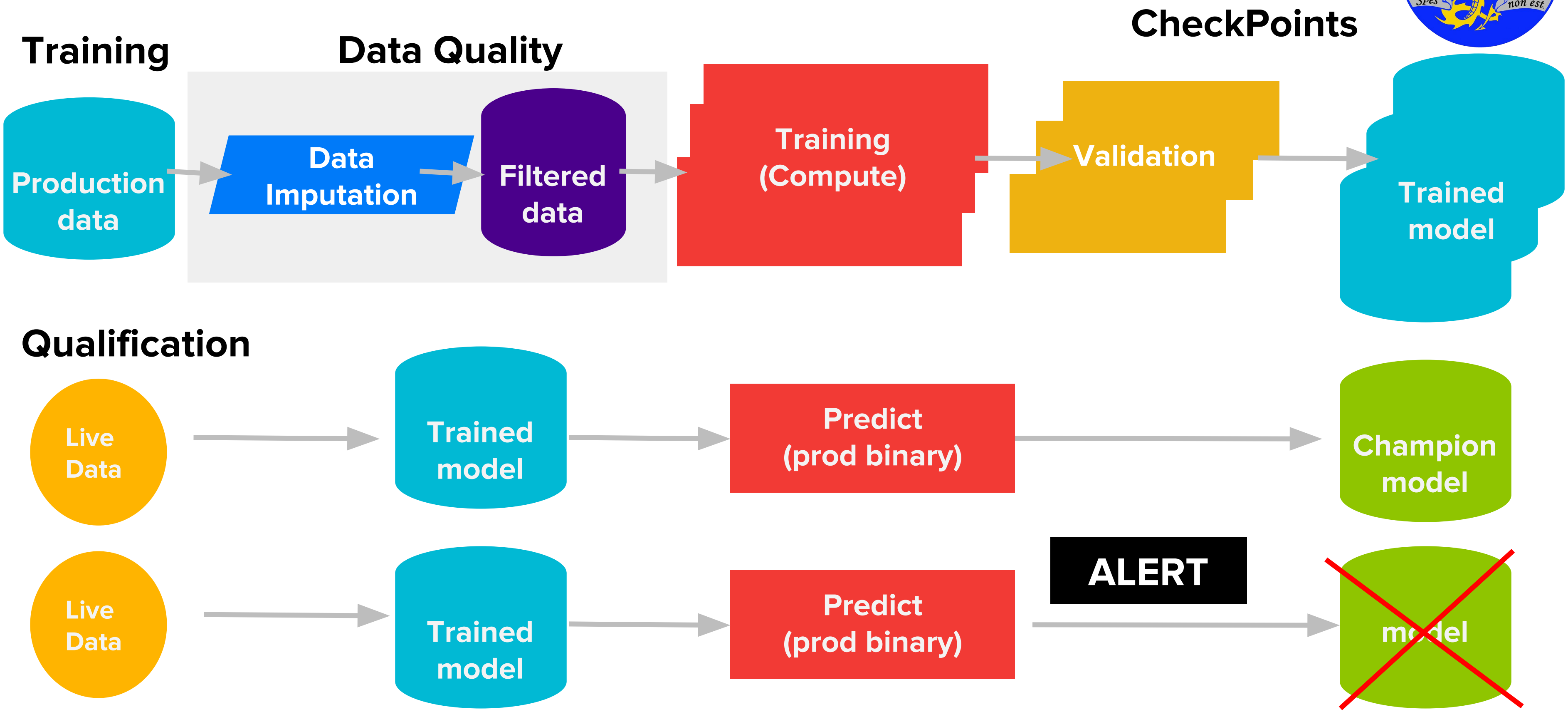
- ❑ Run it in canary
- ❑ Rollbacks must be easy
- ❑ Does a rollback involve human judgement?

¡NO BUENO! :(

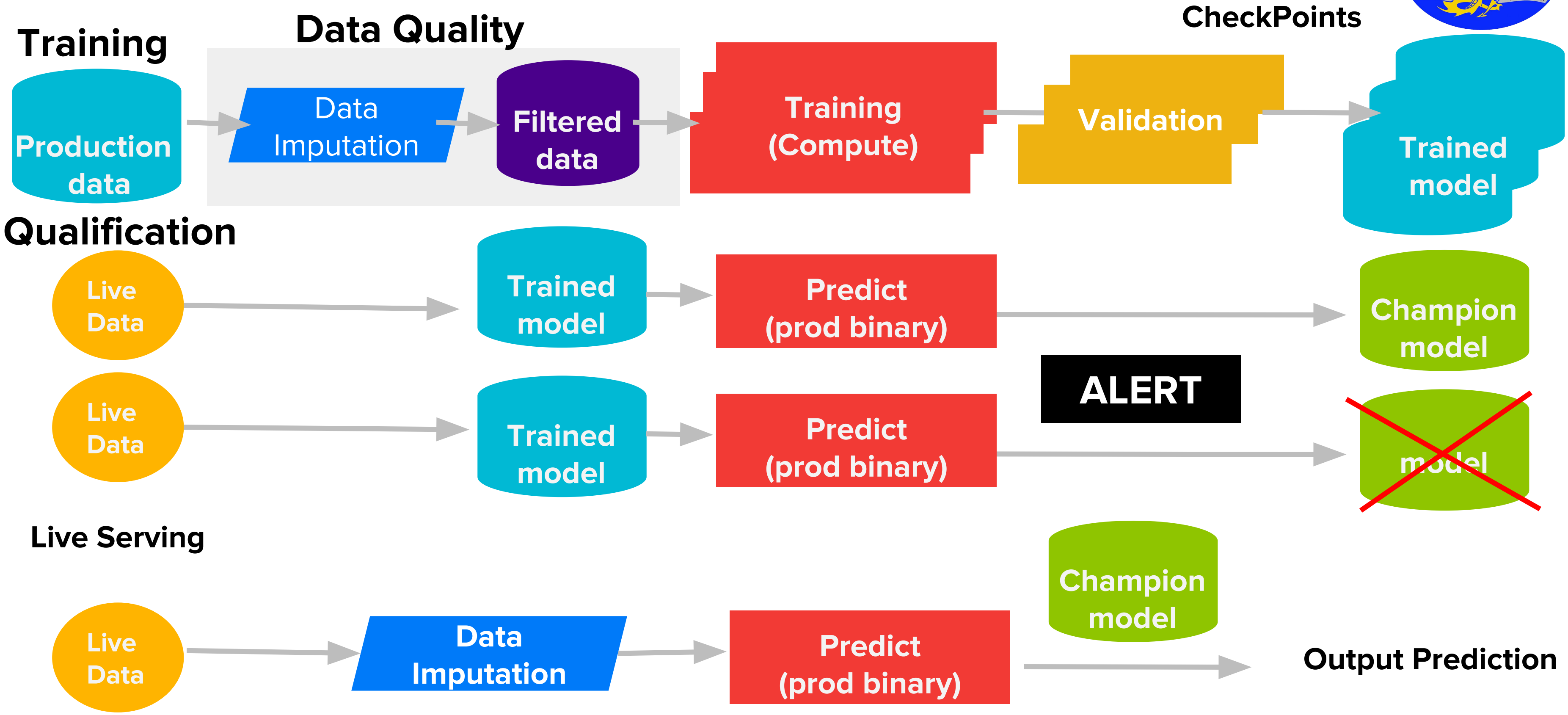
OK Google: What's an ML pipeline like?

De facto production environment (1/2)

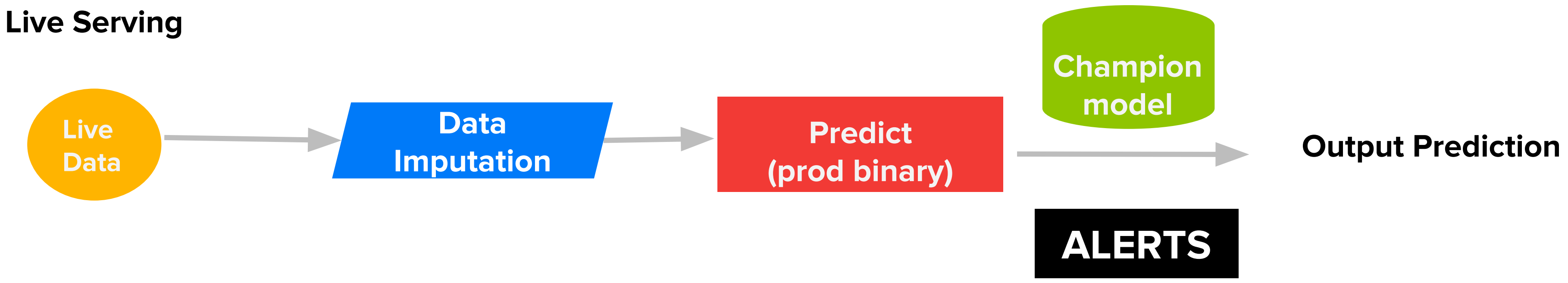
Offline



De facto production environment (2/2)



Monitoring: From SLIs to Alerting



- ❑ Monitor training phase as well as live serving
- ❑ Monitor Prediction latency
  - ❑ Consider TF/RPC overhead
- ❑ Monitor Model aging
  - ❑ Accuracy loss through model aging

What goes wrong when you don't have alerting?

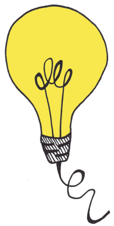
How can you identify this change in behavior?

This is an old story: lack of alerting causes user-facing errors, loss of revenue.

What goes wrong when you don’t have alerting?

How can you identify this change in behavior?

Alerting must be domain-specific



Privacy and Ethics

Privacy in ML

Privacy: When using an individual's data

- **Anonymize** user data
  - Users shouldn't be identifiable from prediction outcomes
- You **must** be able to delete it (remember GDPR)
  - Can you really delete it? How **long** does it take?
  - **Is it automatic?**
- Or Ensure your models **do not have user data** in them--if they do, retrain them as soon as user data is deleted.

Ethics in ML

Ethics in ML

Image source: https://www.flickr.com/photos/70554893@N00/4012154732
licence: https://creativecommons.org/licenses/by-sa/2.0/

Ethics in ML

- Need for external oversight
  - Who can evaluate possible outcomes of the model
- SRE: Be able to ***stop*** ML predictions

Ethics in ML

Experts call for independent oversight, using guidelines from a neutral body.

The AI Now Institute has published its Algorithmic Impact Assessment: https://ainowinstitute.org/aiareport2018.pdf

Conclusions ML Best Practices

Train continuously

Add filtering

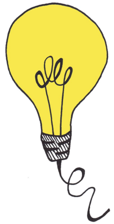
Data imputation

Stamp new models…

… Deprecate old models

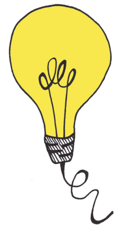
Use domain-specific alerting

Insights that we discussed



- ❏ Migration from previous regression heuristics to ML complicated
    - ❏ The framework changes significantly. No fallback.
    - ❏ Pushing a model is not a simple code change.
- ❏ Training is production
    - ❏ Frequent training (continuously or batch) to push in the order of hours/day.
    - ❏ Training resource demand grows more than prod and requires provisioning.
- ❏ Serving Latency overheads (monitoring)

Insights that we discussed



- Data changes mean problems
  - Monitoring for the data, monitoring for the pipeline: SRE are paged when the separation between the data and pipeline is poor.

  - For example, removing spam content from YouTube: this improves data quality, and leads to better predictions

- Canary relevance: Qualification
- Signatures to prevent models not qualified reaching production.

The Future of ML in Production

- Open source available training data sets
  - Already anonymized + No need to delete user data
- Implications of sharding models
- Dynamically balance load across models *A/B/C*, based on accuracy
- Models as a Service
  - Credit card/Image recognition/Text to Speech as unique APIs

With thanks to

adavies, ademaria, appleton, cfarrar, coppin, dannyp, davebarker, elnota, kozyr, lewinb, lyonya, marcingaw, mdondero, meredithrachel, nfiedel, pafinde, rlb, samg, stross, tmu, xavigonzalvo

And to our colleagues at
Clarifai
ThoughtWorks

With thanks to

very many of our colleagues across Alphabet: DeepMind, Google, YouTube

And to our colleagues at
Clarifai
ThoughtWorks

That's all.

Questions? Comments?

{salim,villavieja}@google.com