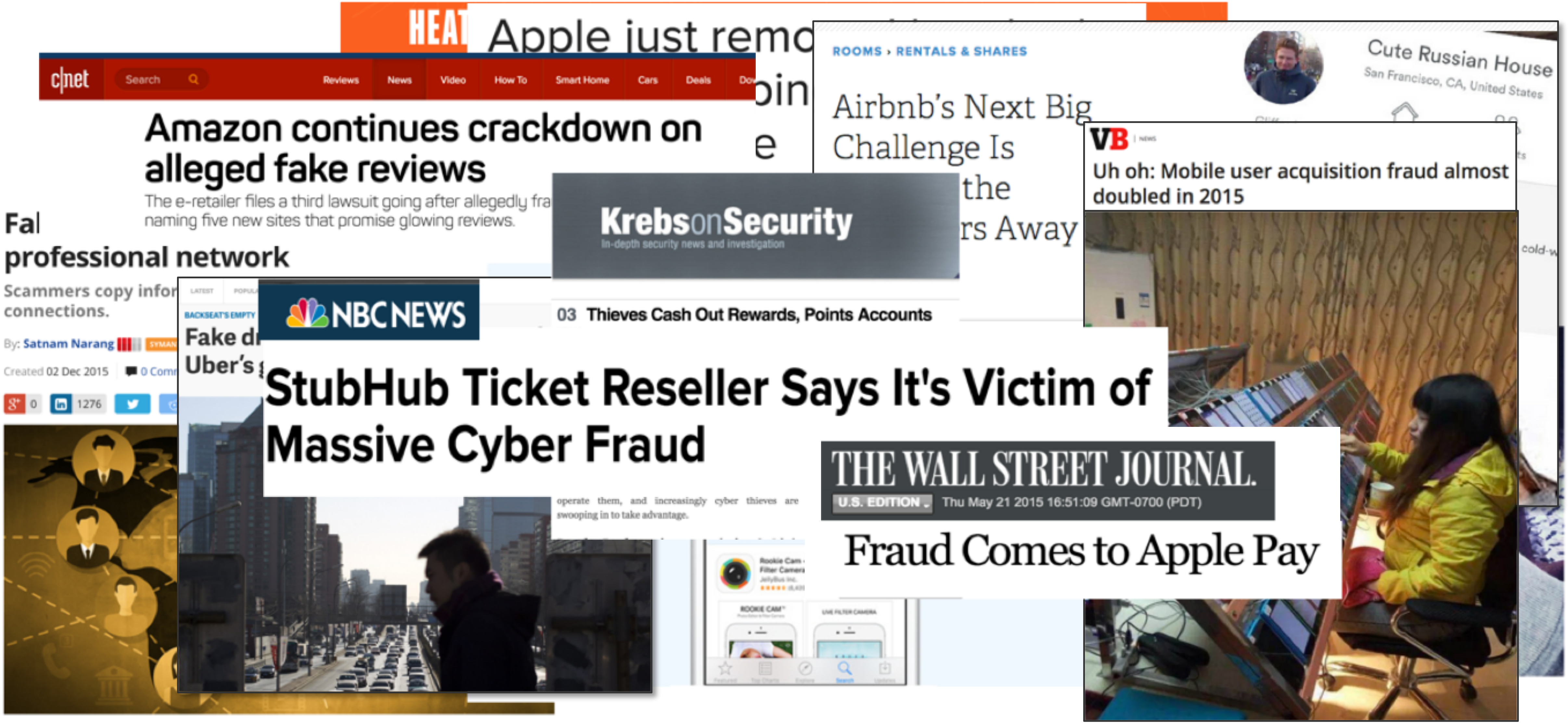
Deep Learning for Large-Scale Fraud Detection

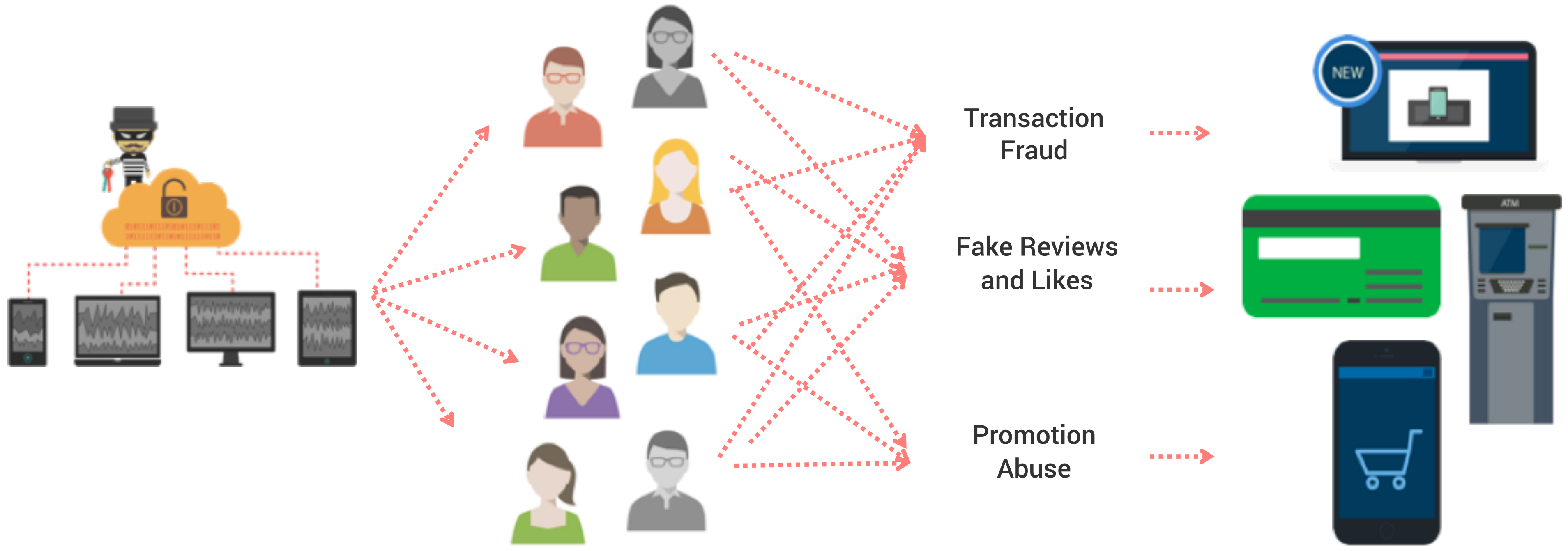
Ting-Fang Yen, Director of Research, DataVisor

Sept 7, 2018

Online Fraud



Modern Attacks are Diverse and Coordinated



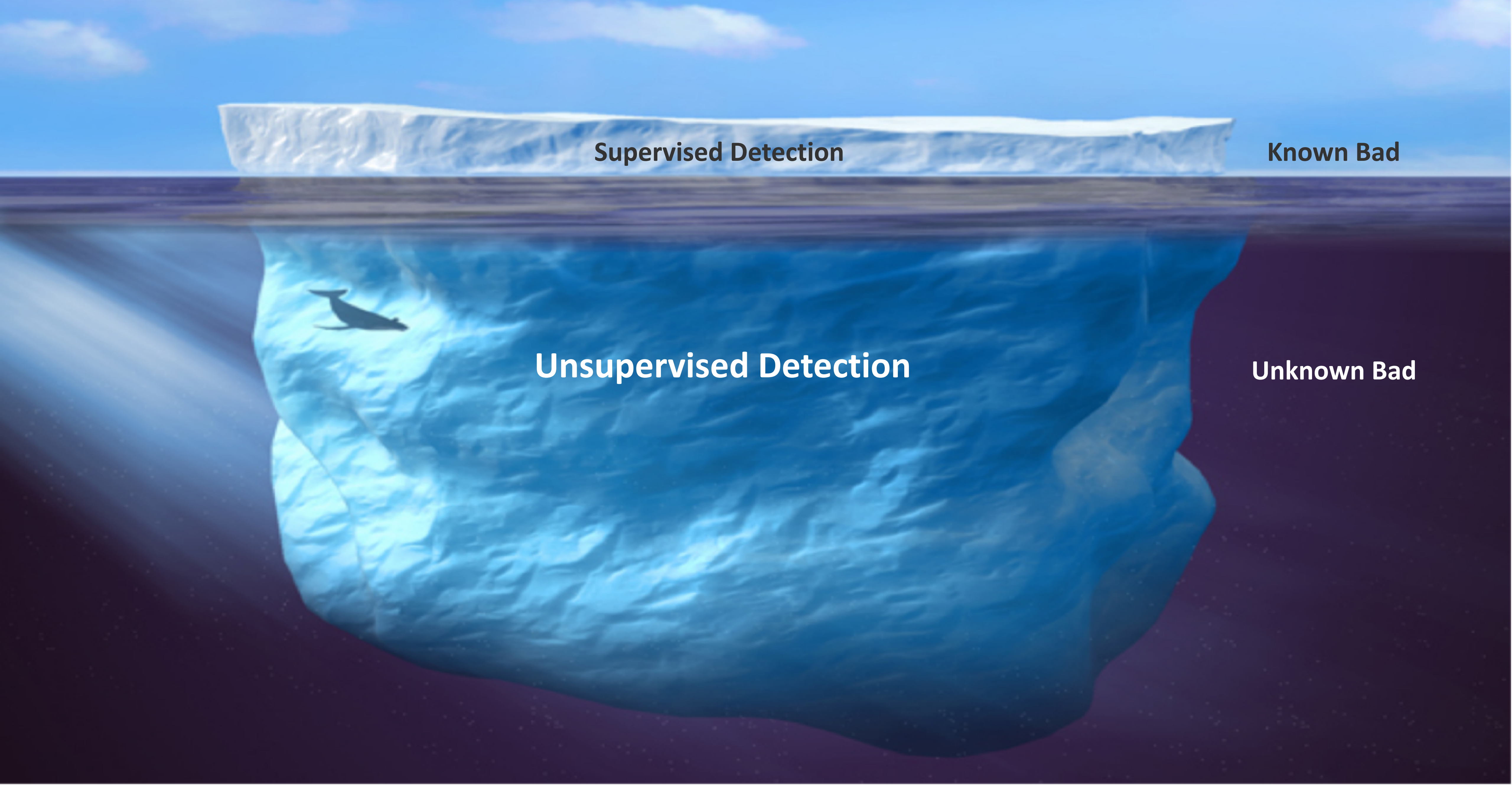
Crime Ring

Malicious Account Army

(Mass Registration and ATO)

Mass Attacks with Diverse Techniques

Vulnerabilities


Supervised Detection
Known Bad
Unsupervised Detection
Unknown Bad

DATAVISOR

Common Digital Info in Application-Level Events

Financial | Social
E-Comm | Mobile

All application-level events from multiple online service verticals

Global Intelligence Network

- 410 Million+ IP addresses
- 3.6 Million+ Email domains
- 300,000+ OS versions
- 5.3 Million+ User agent strings
- 160,000+ Device types
- 700,000+ Phone prefixes
- ……

From **3 Billion+** global users, **600 Billion+** events and growing

Leverage Combined Data

Global Intelligence Network

Financial | Social

E-Comm | Mobile

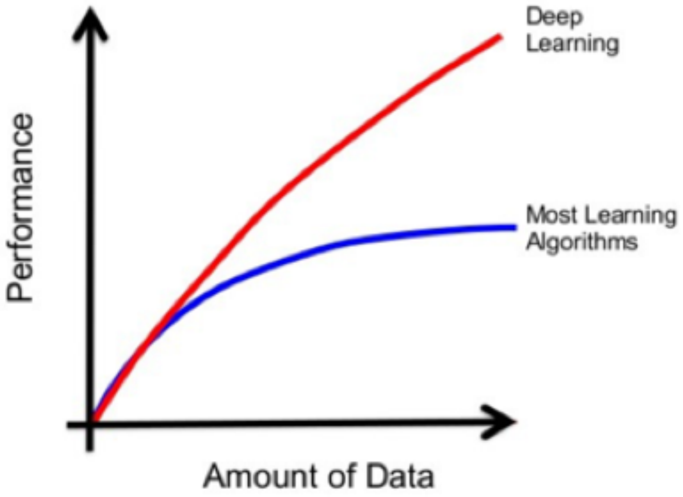
Derive granular user behavior information

- New user ratio
- Fraudulent user ratio
- First/Last seen time
- Proxy/Data center IP
- Geolocation
- … …

Deep Learning

Fraud score

Deep Learning Has Seen Tremendous Success



Many Deep Learning Tools

mxnet

Microsoft CNTK

theano

DL4J
DEEPLEARNING4J

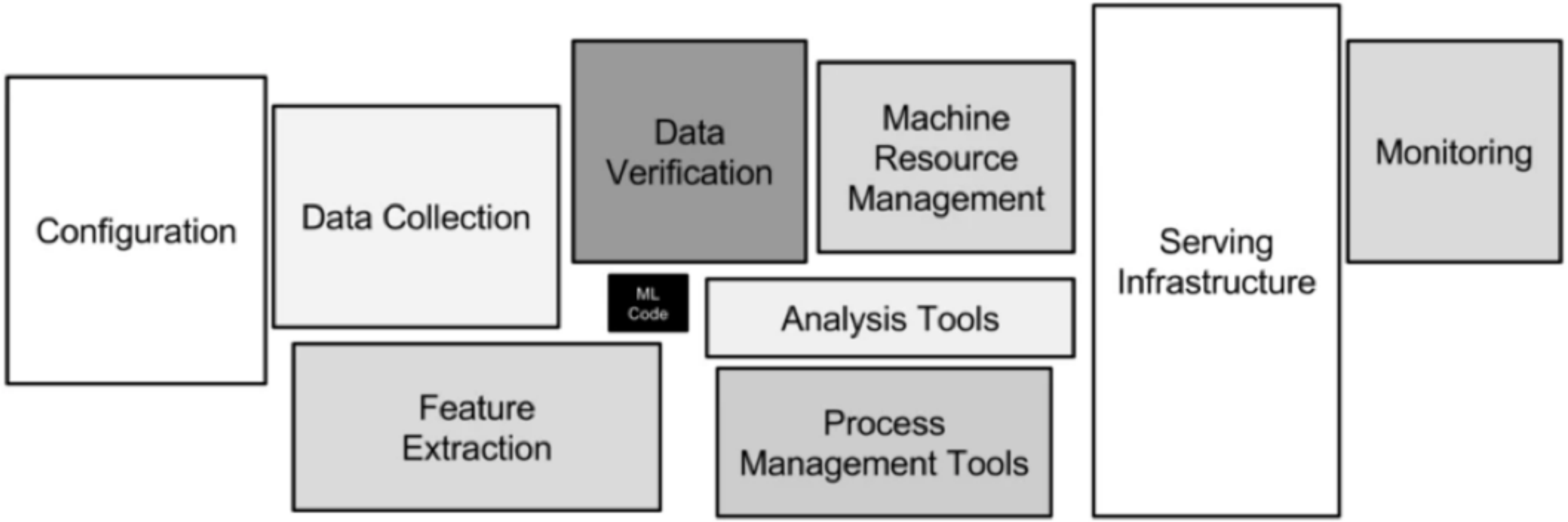
PYTORCH

Chainer

Caffe2

K Keras

Serving ML/DL in Production is Challenging



"The required surrounding infrastructure is vast and complex."

Sculley, D., et al. "Hidden technical debt in machine learning systems." *NIPS*. 2015.

Spark and TensorFlow's Strengths in Productionizing Machine Learning

Pros:
- Unified engine (end-to-end solution)
- Simple API
- Speed

Cons:
- Deep learning integration under development

Pros:
- Production ready (if done right)
- Extensive ML API for various tasks

Cons:
- Limited data pre-processing support
- Not end-to-end solution

Combining Spark and TensorFlow for DL Tasks

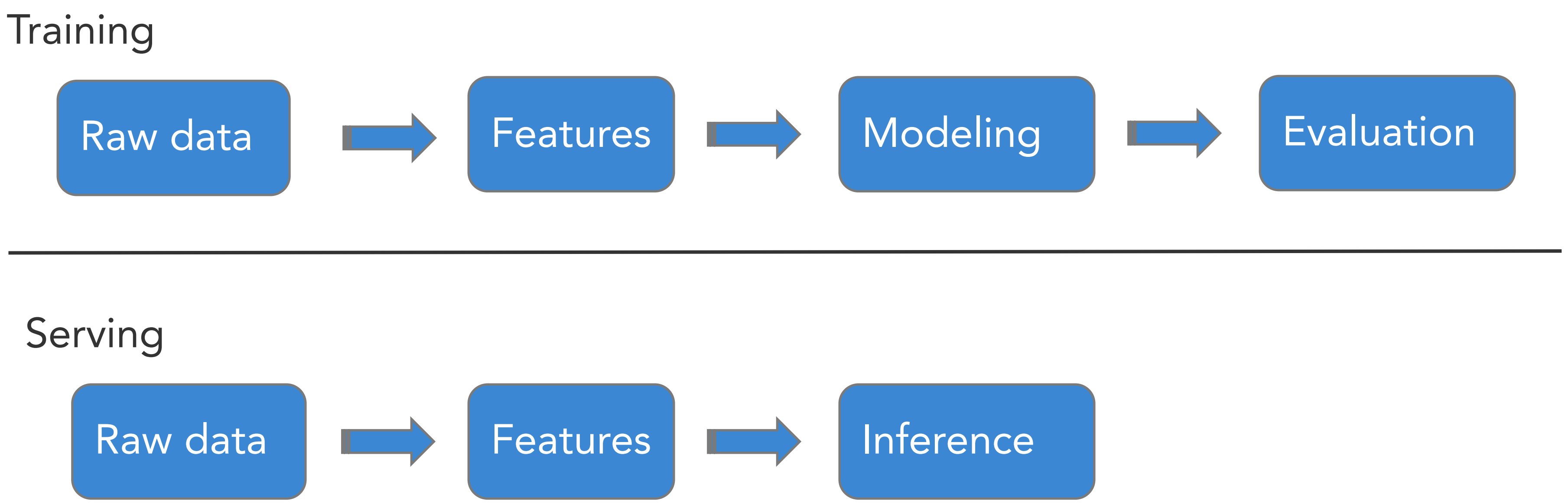
Active direction in Spark community

Spark integration with TensorFlow

- TensorflowOnSpark
- DeepLearningPipeline
- Tensorframe
- … …

Can we combine TensorFlow with Spark applications (Java) to apply DL to fraud detection?

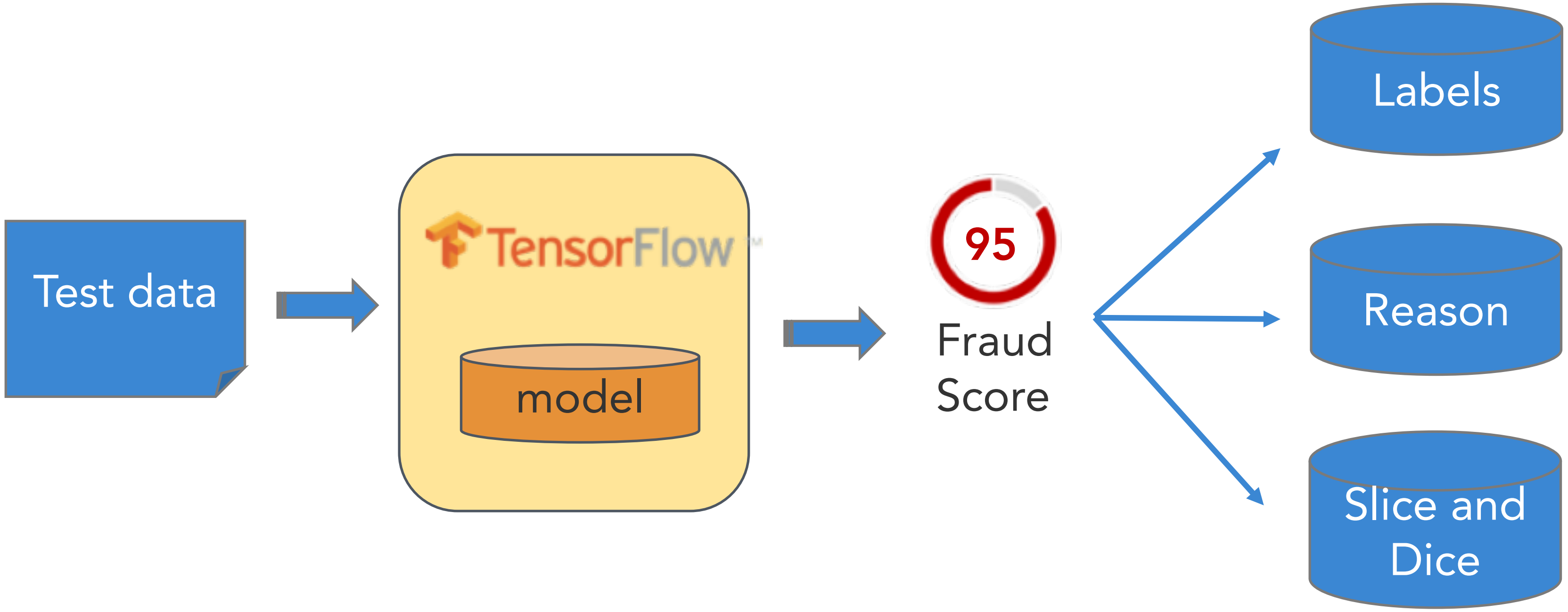
A Typical Machine Learning Workflow



Requirements:

1. Treat anti-fraud application as a classification problem
2. Training serving consistency
3. Post analysis to understand model performance in new applications

Understanding Inference Results is Critical



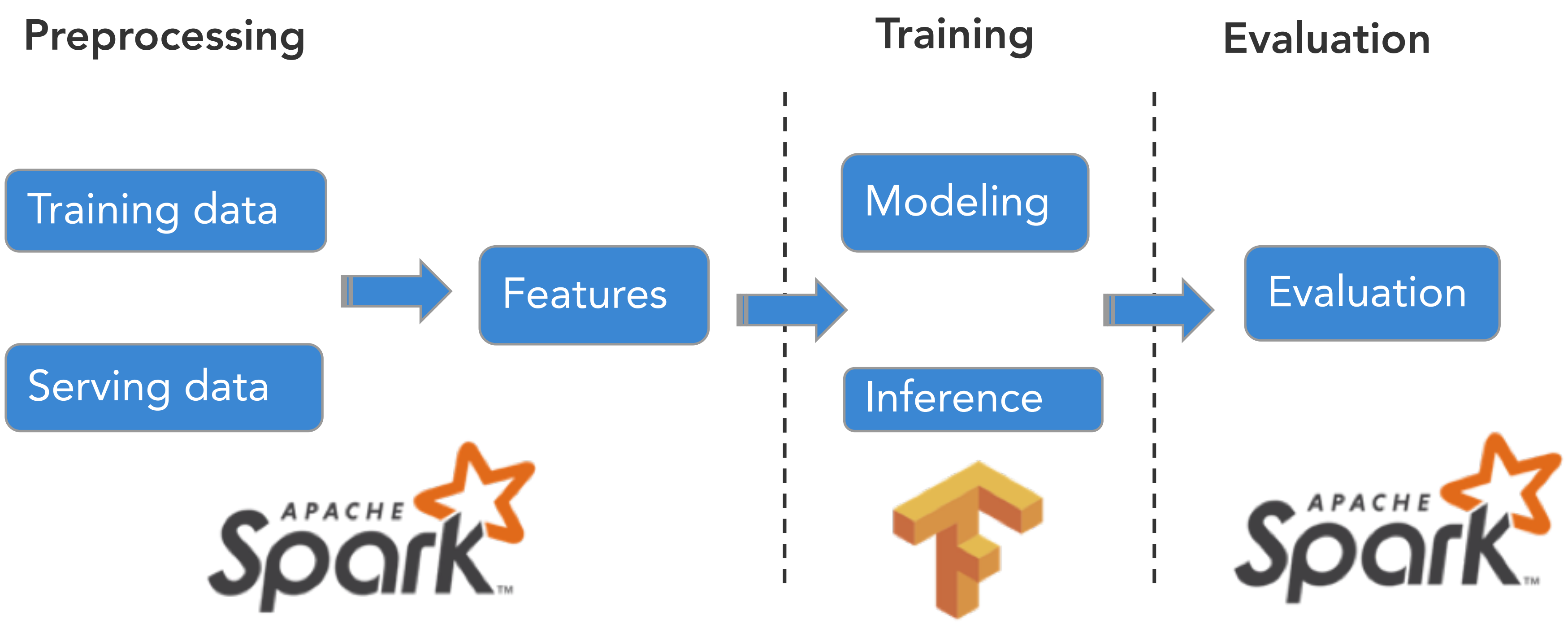
- Source of label

- Top features
- Heuristics

- Imbalanced labels
- Subsetting data
- FN/FP debug

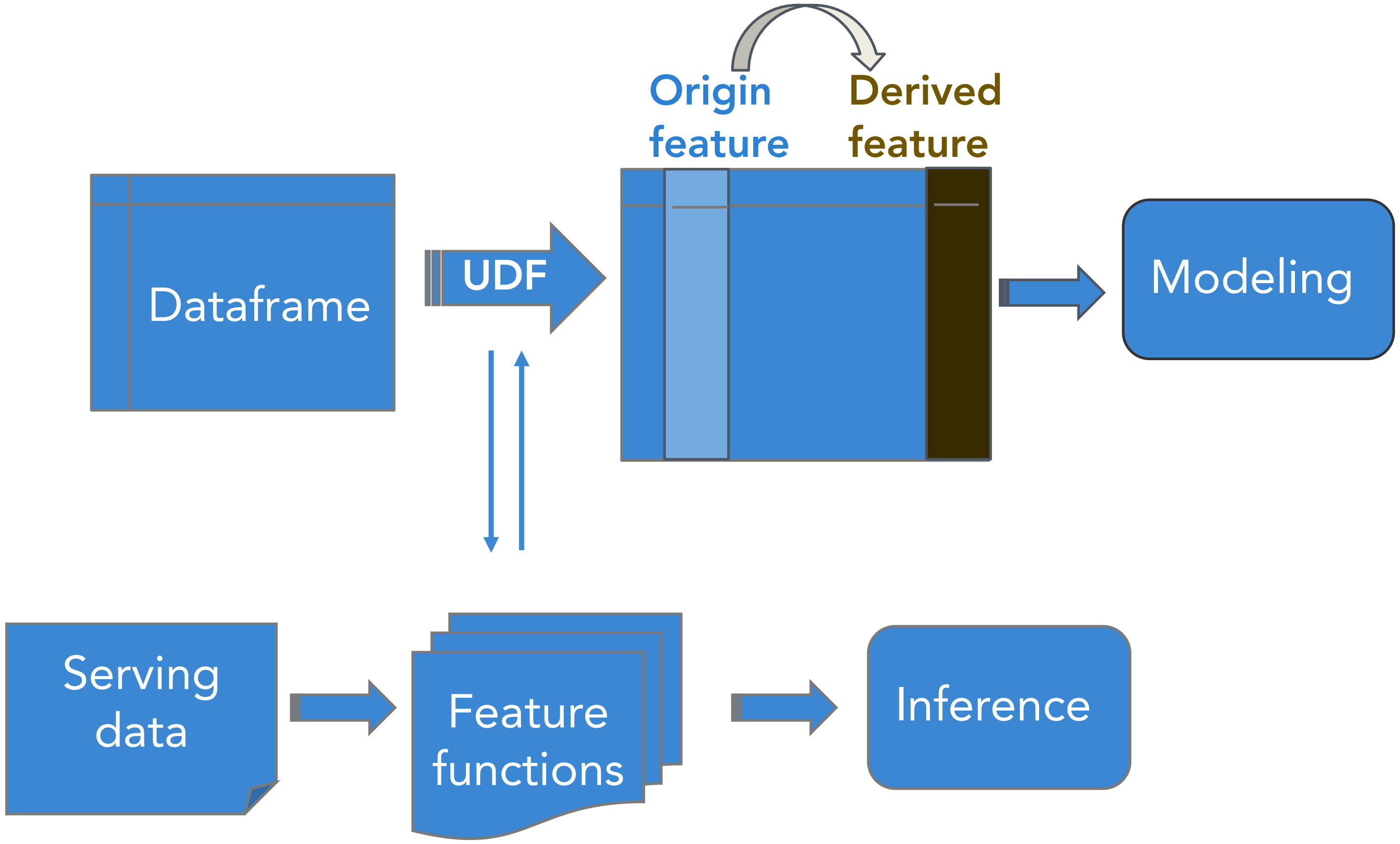
Being able to post-analyze on the inferred data is critical for model development

How We Leverage Spark and TensorFlow



- Spark is used in pre-processing pipeline to prepare training data for DL models
- TensorFlow is used to handle DL model training and serving workflows
- Spark is used for post-analysis and model evaluation

Spark for Generating Training and Serving Data



Pre-processing

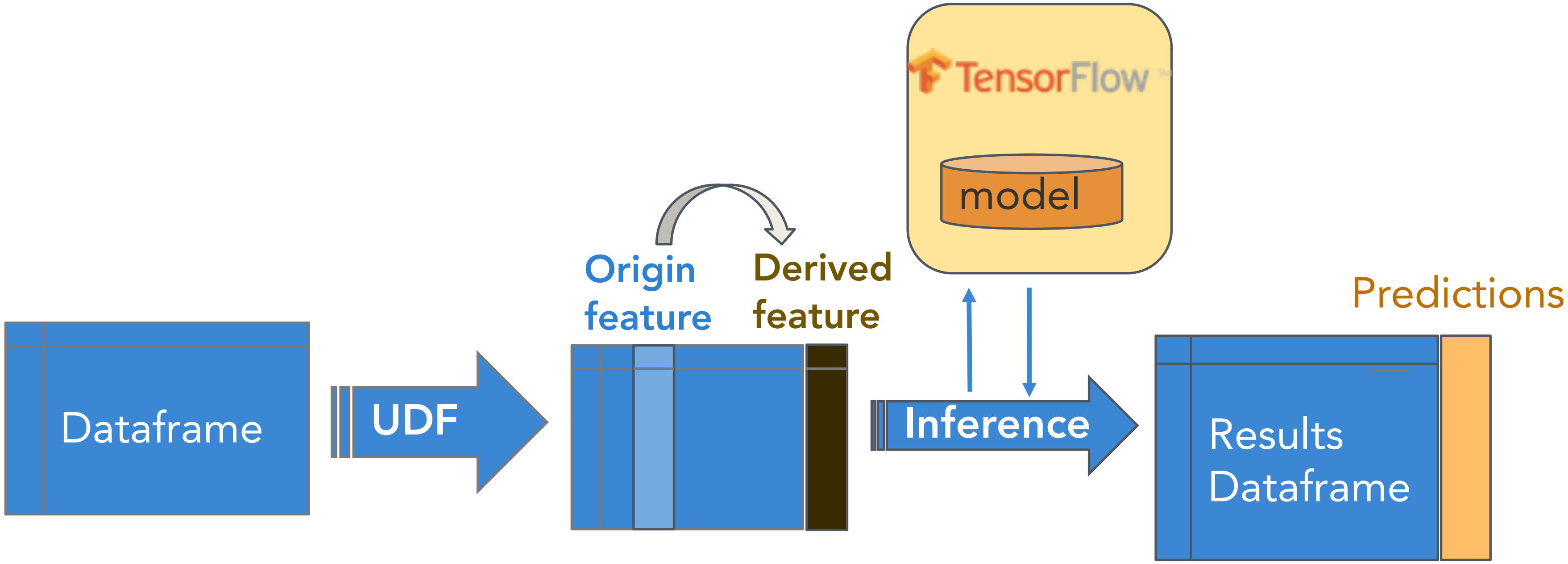
- Load data into dataframe
- Each user defined function (UDF) is built from a feature function
- Uniform API

Serving

- Every entry of data point is pre-processed and then fed to DL model for inference
- The same feature function is used to process data at serving time

DATAVISOR

SparkSQL and TensorFlow Serving for Batch Inference



- Use dataframe API to load random testing data
- Leverage TF Serving for model inference
- Developed client (Java) to "talk" to TF model
- Inference result is returned as a new dataframe with one extra column – "predictions"
- Post-analysis can be done on the resulting dataframe – with both features and scores

Leverage Combined Data

Global
Intelligence
Network

Financial	Social
E-Comm	Mobile

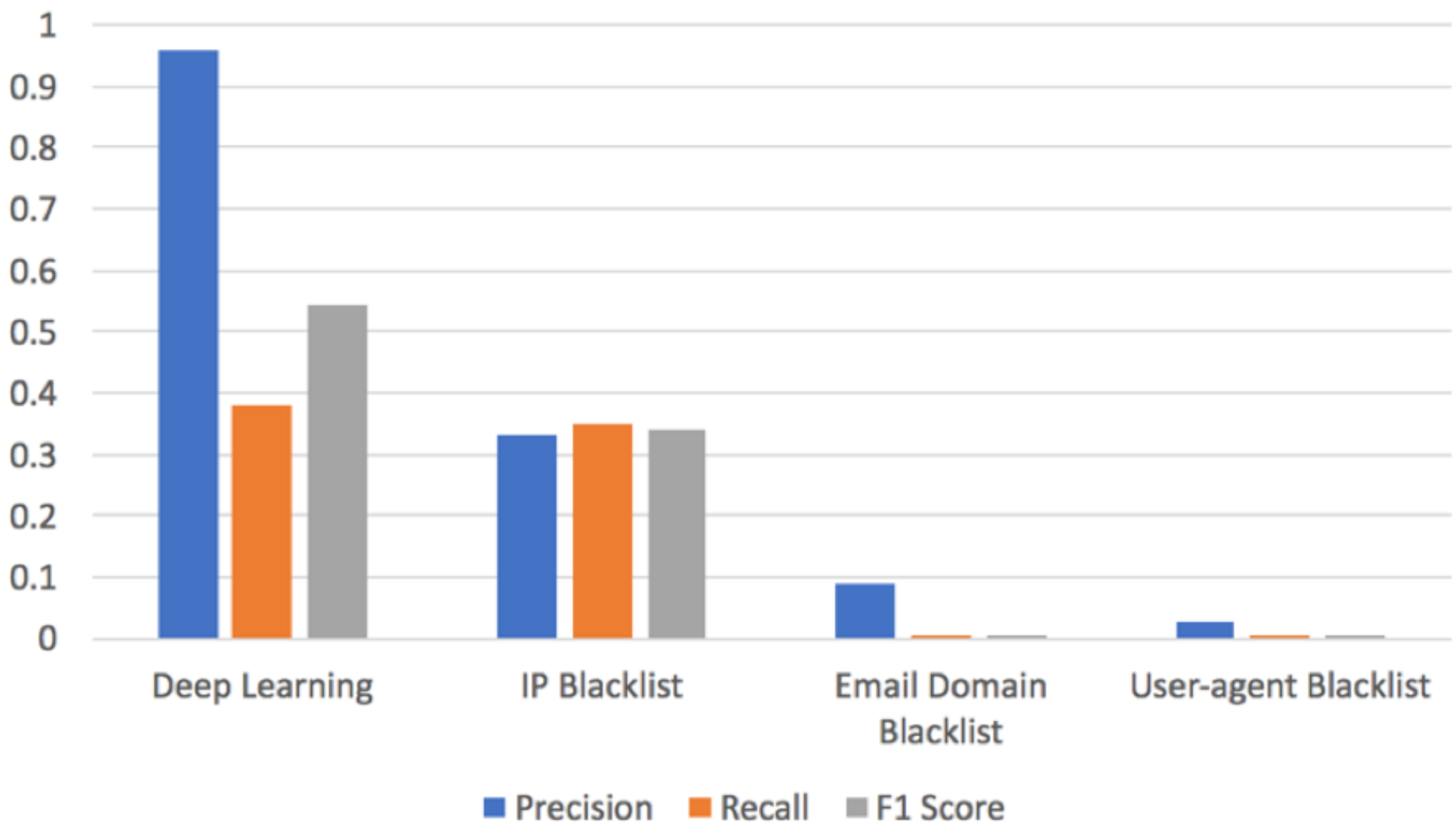
Deep Learning

Fraud score

Derive granular user behavior information

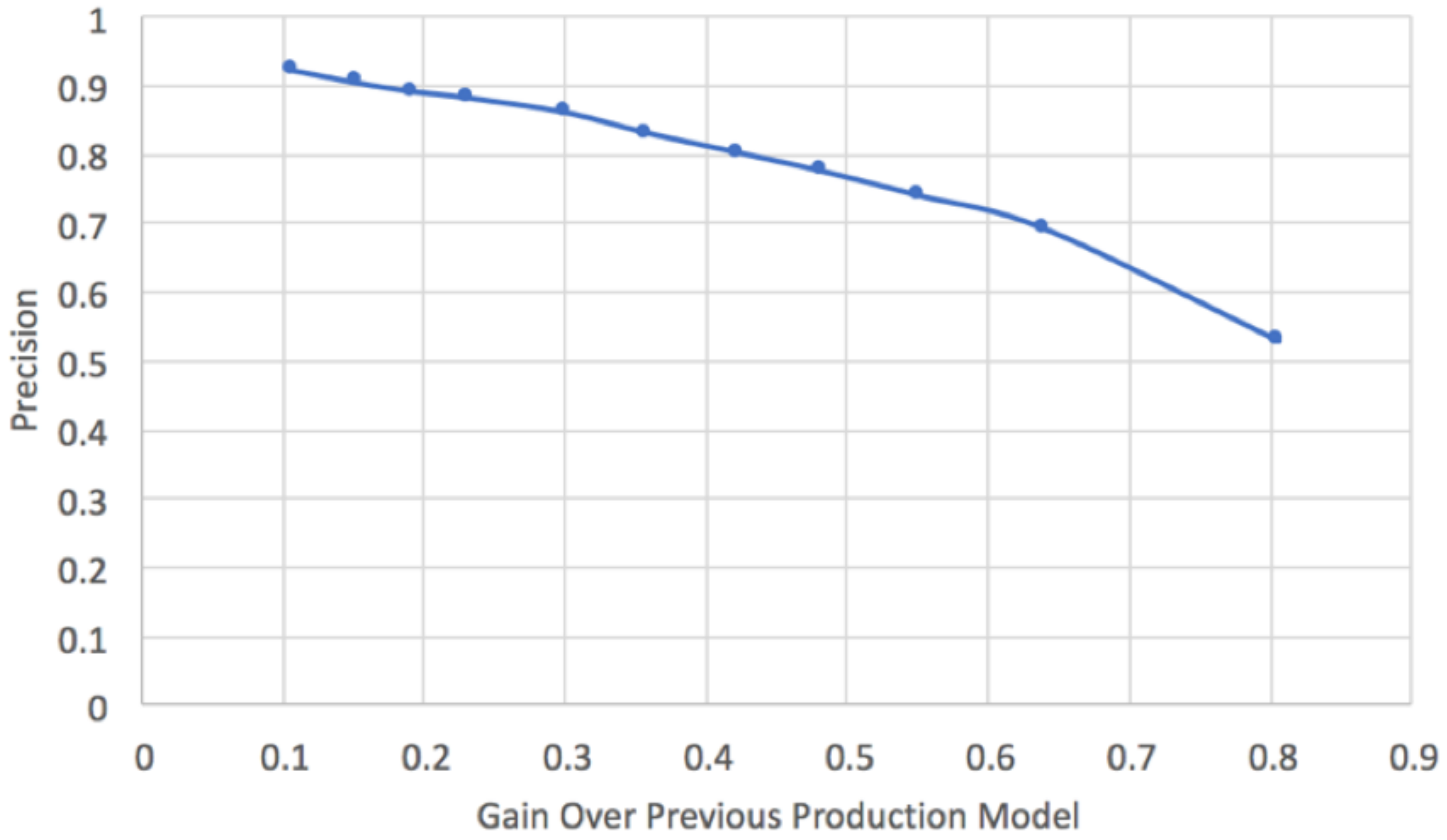
- New user ratio
- Fraudulent user ratio
- First/Last seen time
- Proxy/Data center IP
- Geolocation
- … …

Evaluation: Multiple vs. Single Dimension

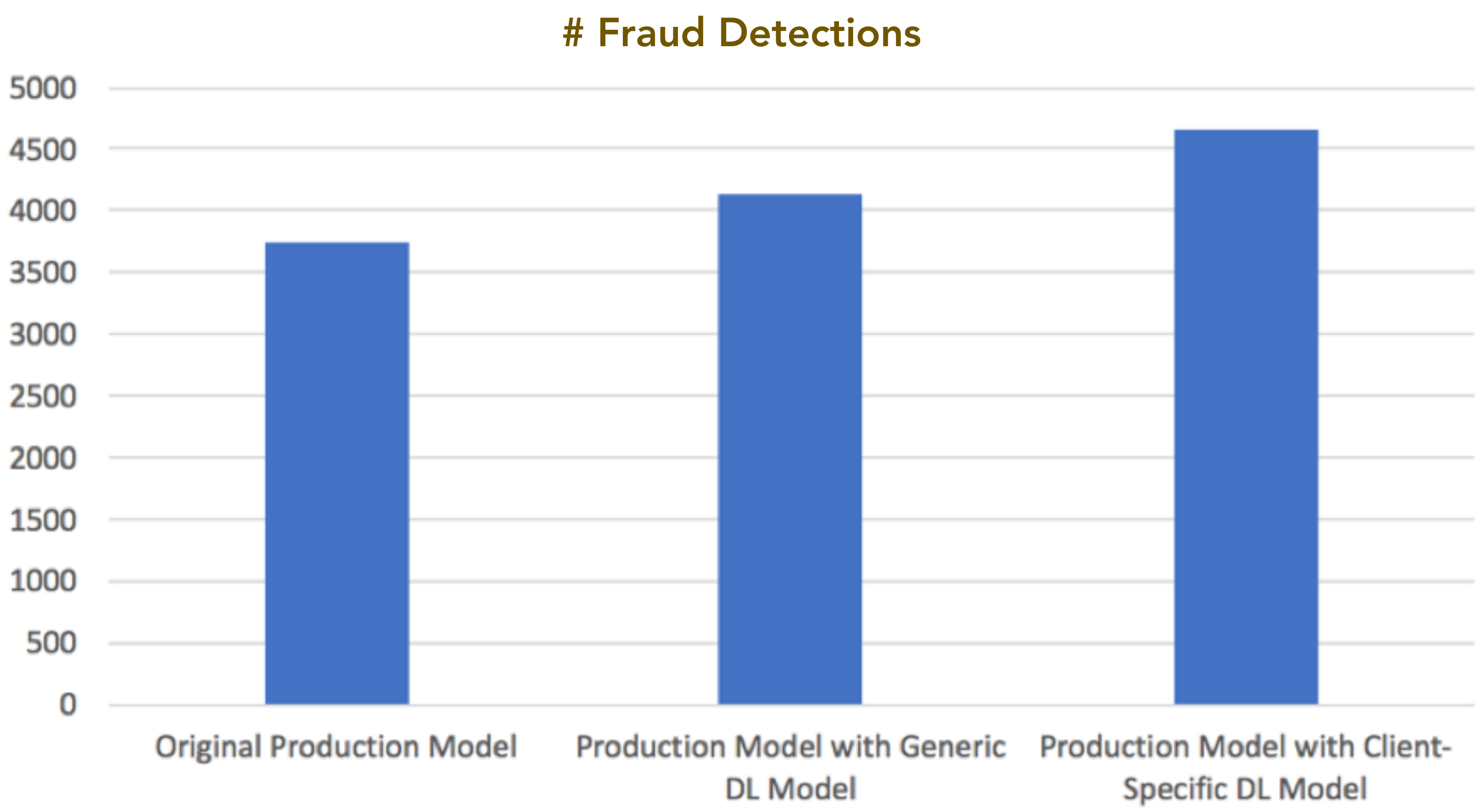


DATAVISOR

Improving Existing Production Model



Improving with Client-Specific Labels



DATAVISOR

Summary

- Spark + TensorFlow makes deep learning applications simpler
- Lessons from applying ML to new domain
  - Feature engineering
  - Post analysis to understand results
- Deployed successfully in production clients with improved results

- Thanks to Arthur Meng, Yang Xu, Yuetong Wang, Boduo Li