

The Rapidly Changing Art of Computer Vision Development

 occipital

Denver Dash, Director of Machine Learning

DevPro Conference 2019

San Francisco, CA

Boulder, CO

Gainesville, FL

Madrid, Spain

THE TEAM



50+ strong

9 former founders

3 original Kinect hackers

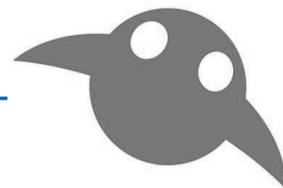
KEY BACKERS



ACQUISITIONS



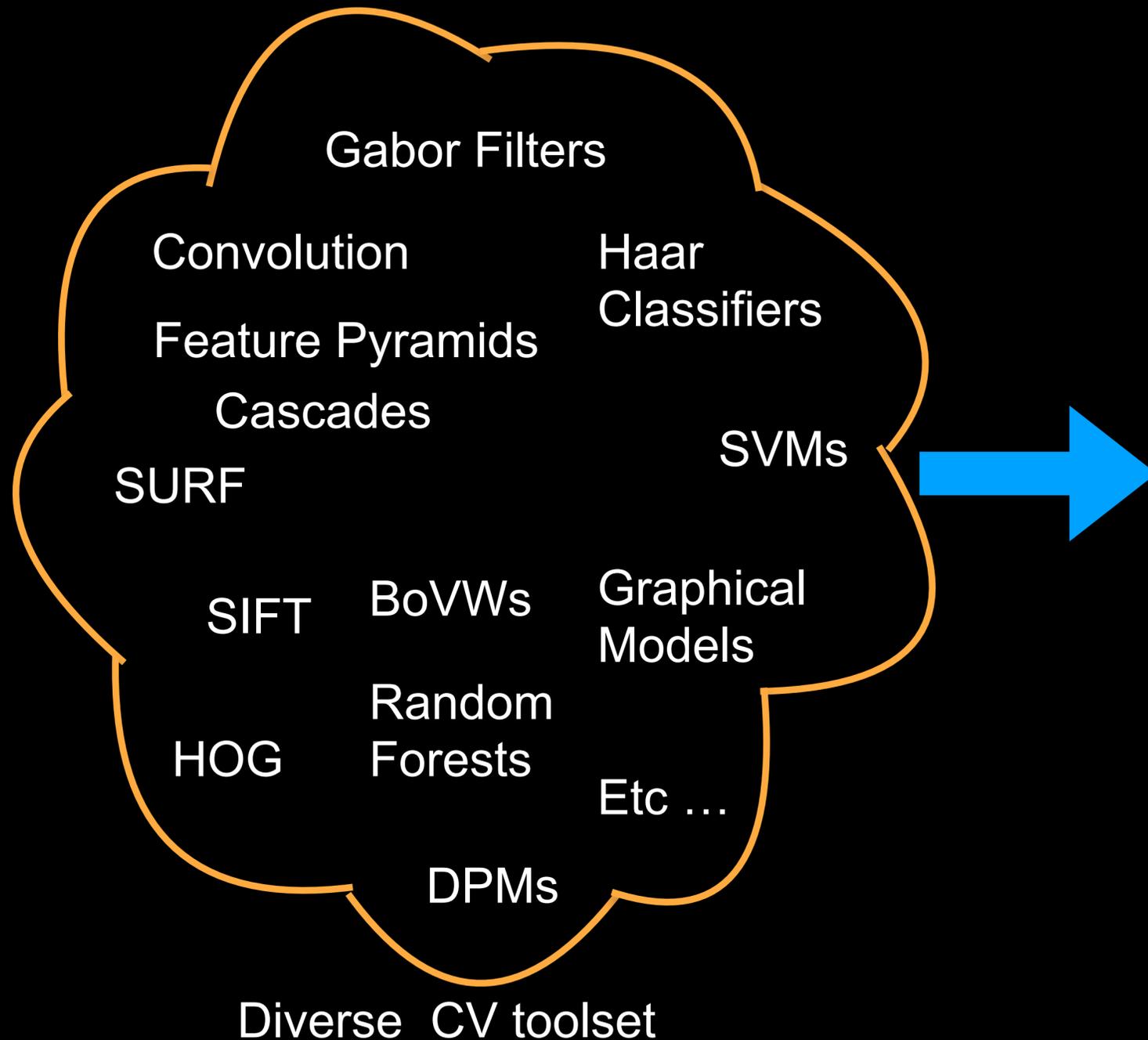
TEAM PAST WORK



Microsoft HoloLens

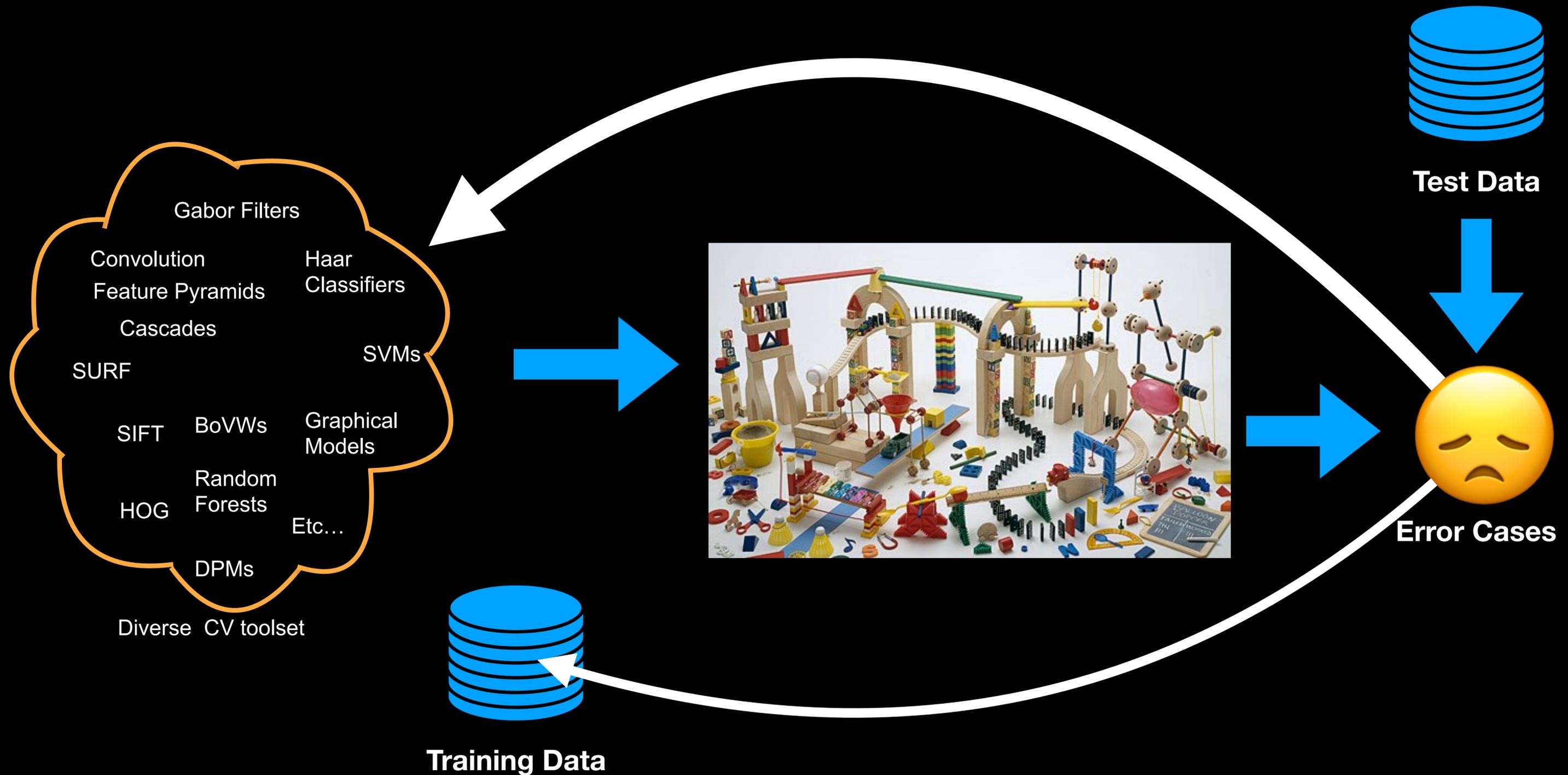


The Practice of Semantic CV for Industrial Use (<2014)

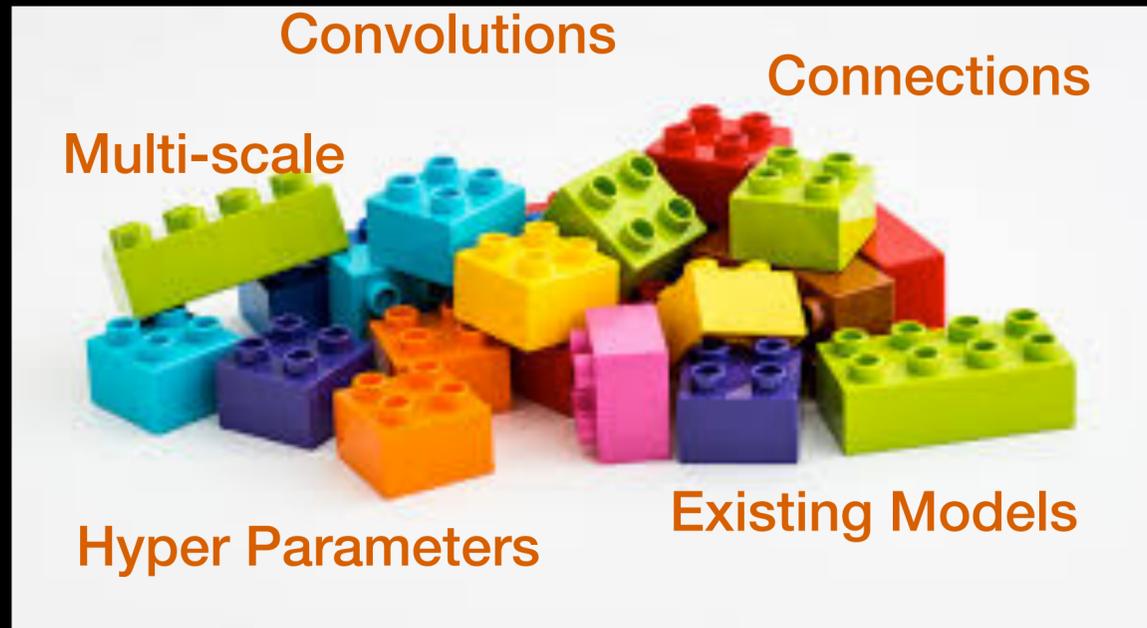


Industrial System Composed of Existing Techniques Carefully Glued Together

The Practice of Semantic CV for Industrial Use (<2014)



The Practice of Semantic CV for Industrial Use (>2014)

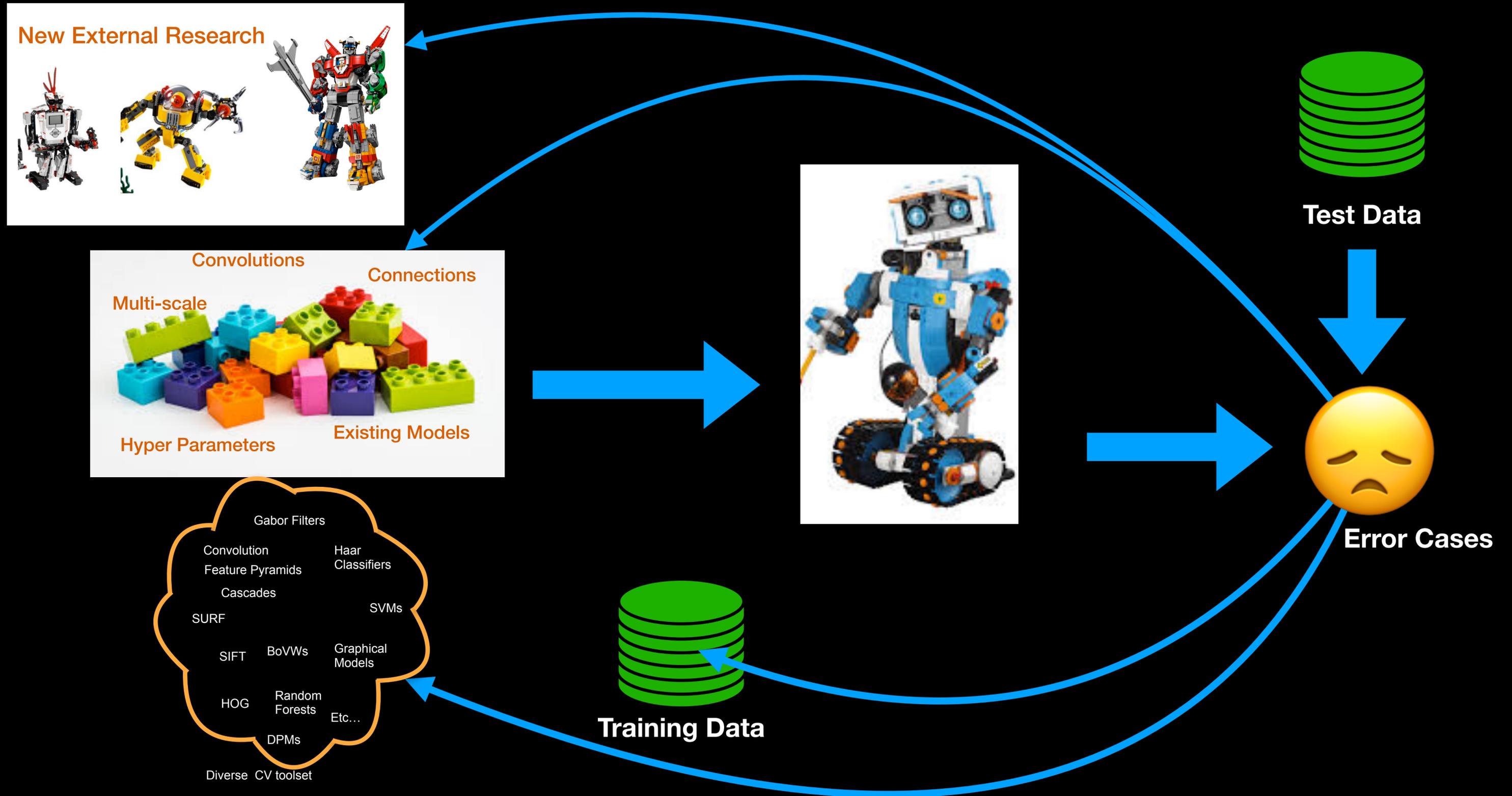


More Uniform toolset

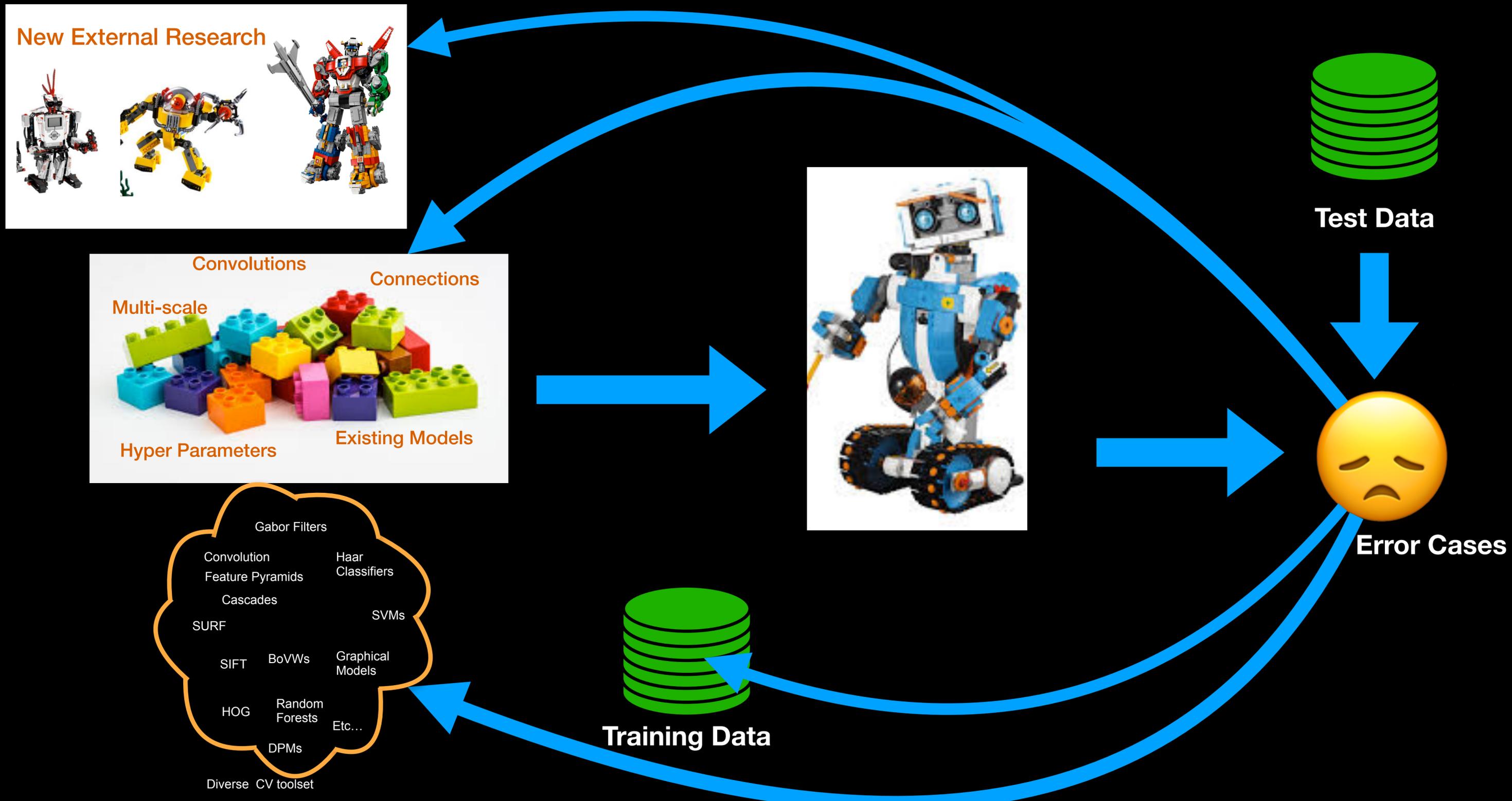


Modern Industrial CV Model

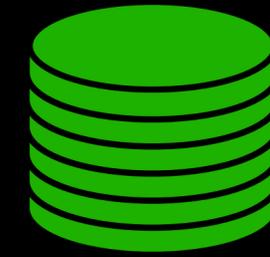
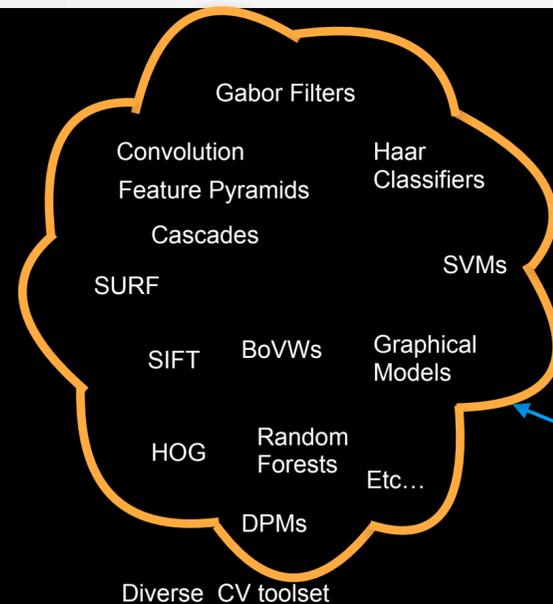
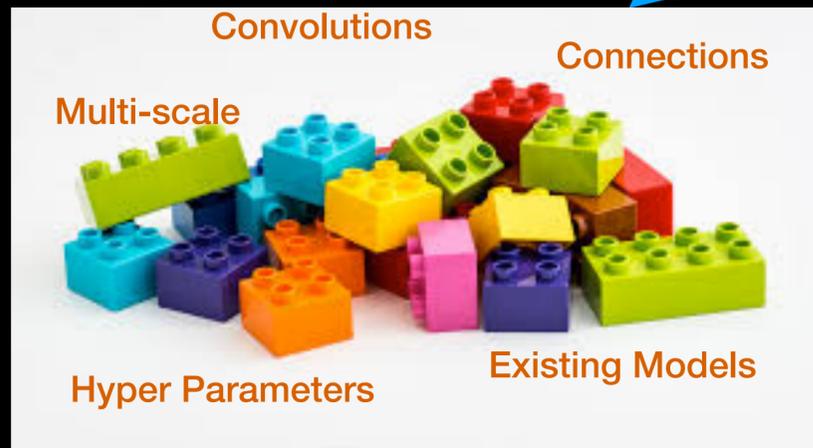
The Practice of Semantic CV for Industrial Use (<2014)



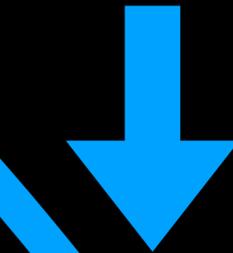
Priority Order for Development (Typical)



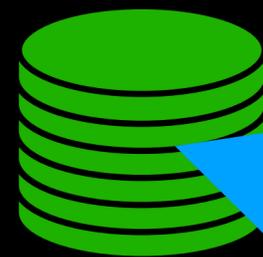
Priority Order for Development (Mine)



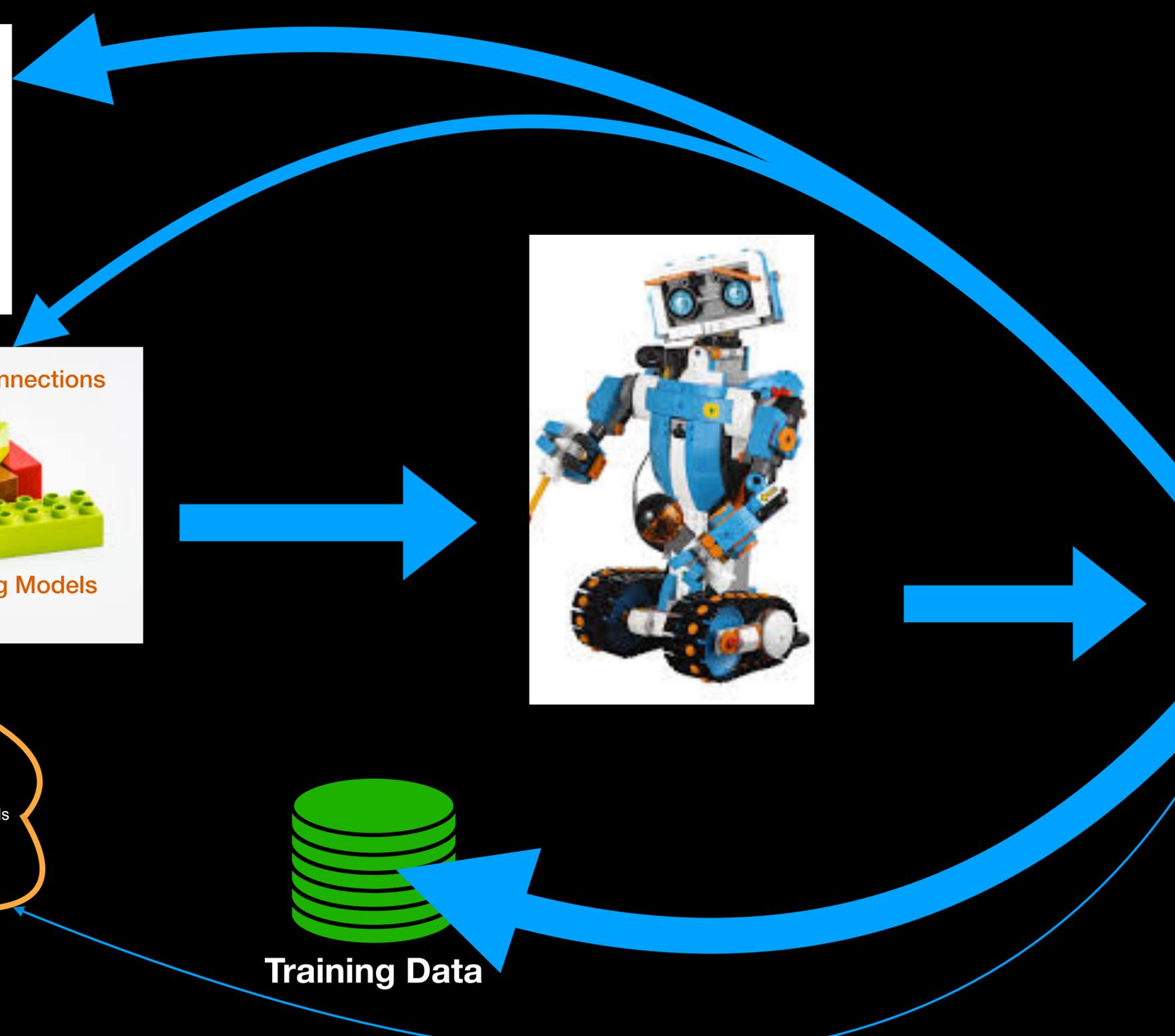
Test Data



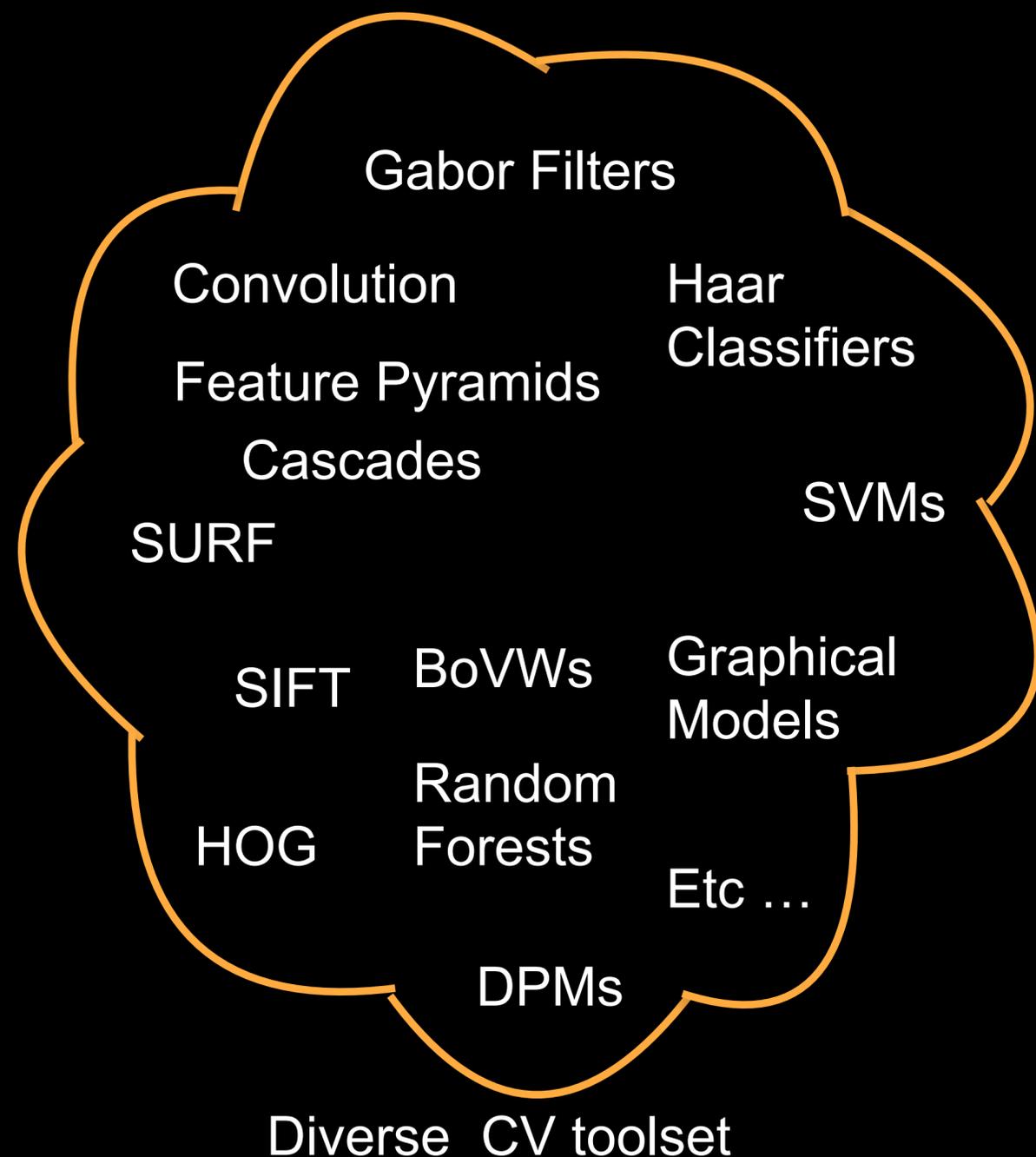
Error Cases



Training Data

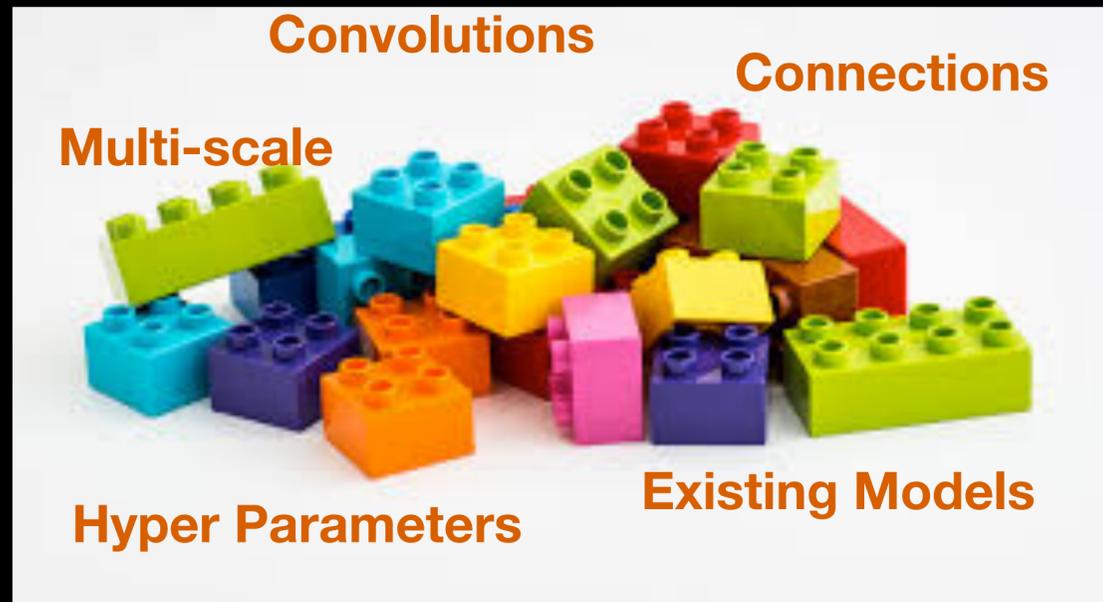


Priority 4: The 90's Toolbox



- Resist the temptation at almost all costs
- Adding these increase complexity of the system substantially and make your code more error-prone
- As a last resort can be used as a “bandage” on an “almost working” system that you need to deploy urgently
- Treat these changes as “technical debt” and work to pay it back ASAP.

Priority 3: Tweaking Model Hyperparameters



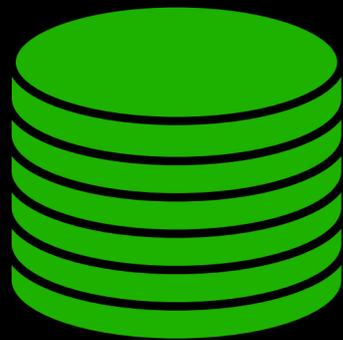
- Do this when you are waiting for more labeled data and have no viable new research that could be tested
- Don't start this activity in earnest until you have quantitative metrics implemented so you can see the effects of your architecture tweaks
- Massively vary hyperparameters and pick the best one quantitatively
- Don't get too attached to any particular architecture
- Don't expect a huge breakthrough but you might get incremental gains that add up over the long term
- Don't overuse your test set during this process - rely on a validation set instead

Priority 2: New External Research



- Take advantage of a huge and active research community
- Follow on Twitter academic leaders doing research in your subject
- Schedule weekly or semi-weekly code reviews to test out new methods on your data
- Schedule comprehensive reviews of the relevant work at NIPS, CVPR, ECCV, ICLR, etc and test the code that is released around the time of the conference
- Usually will not be “game changers” without some tweaking (but sometimes they will be)

Priority 1: Data



Training Data

- The most important way to improve your model is to train on the right data
- Invest heavily and early in data infrastructure
- Think carefully about clever ways to augment your data
- Keep a characteristic training and testing set for your problem and use it to fine-tune new models
- Develop quantitative metrics early on and use them to drive progress
- Inspect error cases and retrain (hard negative mining) and collect more training data similar to those

Case Study: Hand Tracking @ Magic Leap



tónandi

studios

IN COLLABORATION WITH

sigur rós

2019 D.I.C.E. Award Winner for Immersive Reality Technical Achievement



Photography: Ieva Aust

“ The Tónandi experience is more like hiking or scuba-diving in your house while also being surrounded by supernatural beings. It’s appealingly disorienting. By the end, orchestration is sizzling — I can almost feel it—through my fingertips. When it’s over, I ask if I can go again. ”

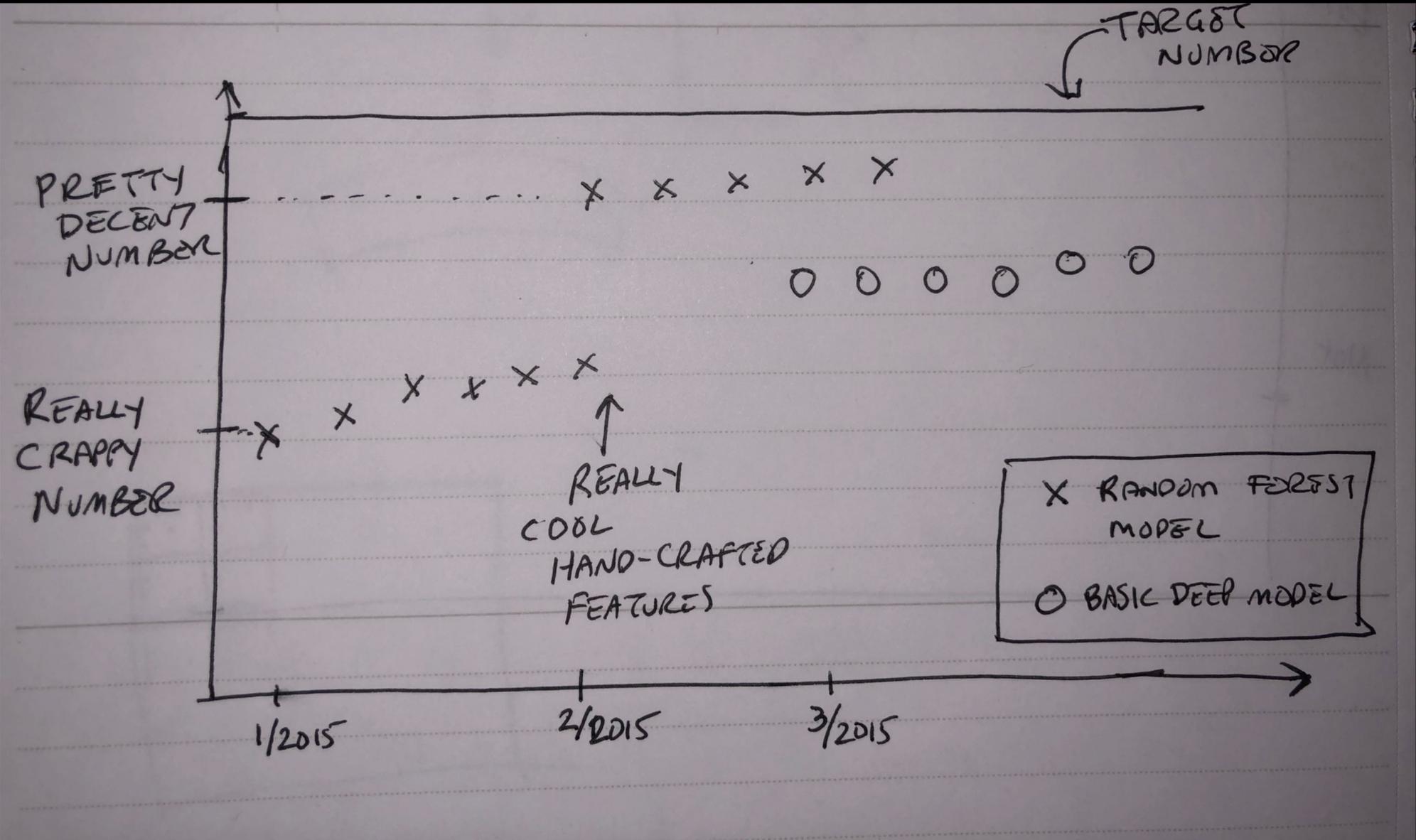
Marc Hogan - Pitchfork

SHOT IN ENGINE

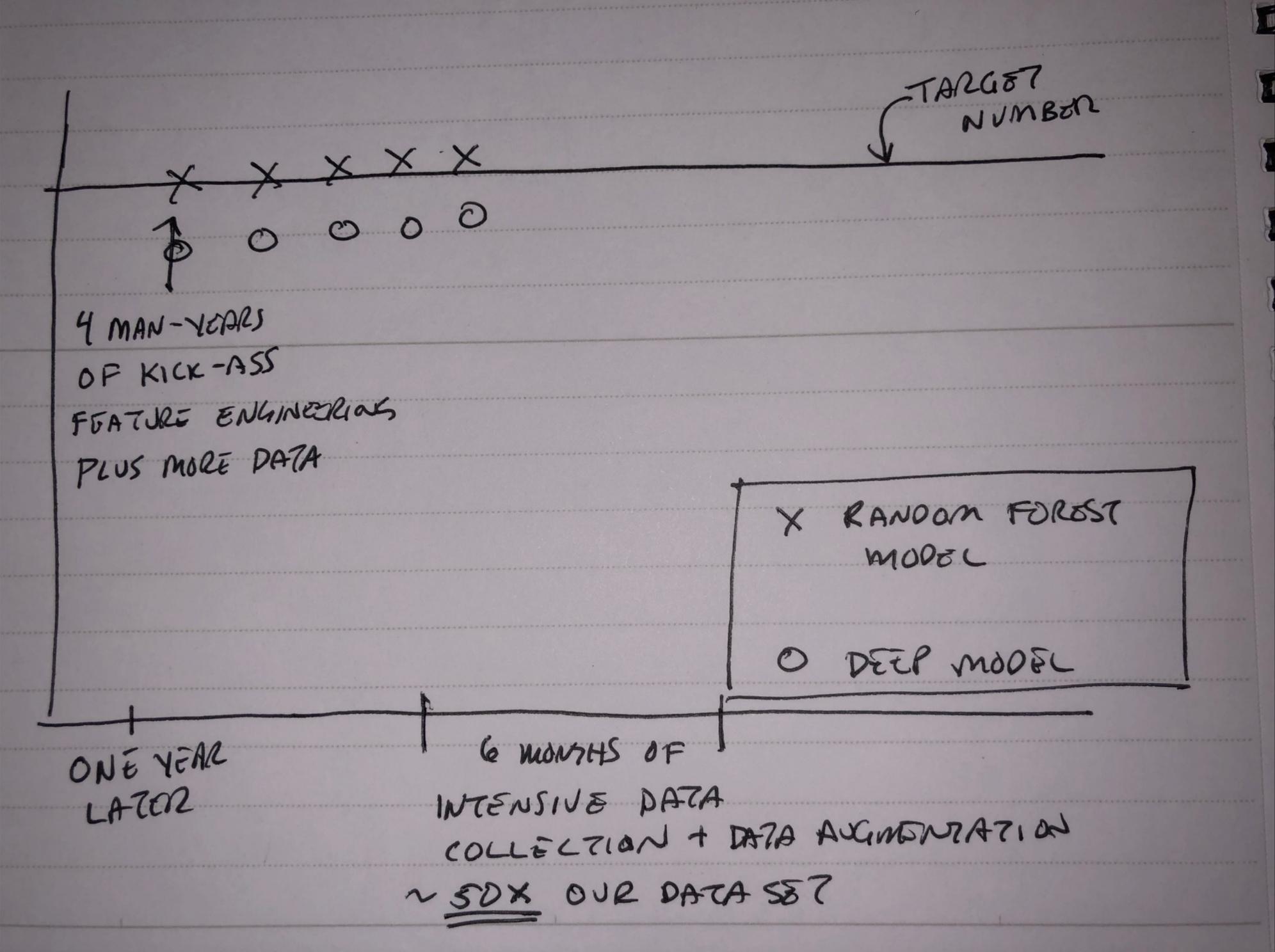
<https://www.magicleap.com/experiences/tonandi>

Case Study: Hand Tracking @ Magic Leap

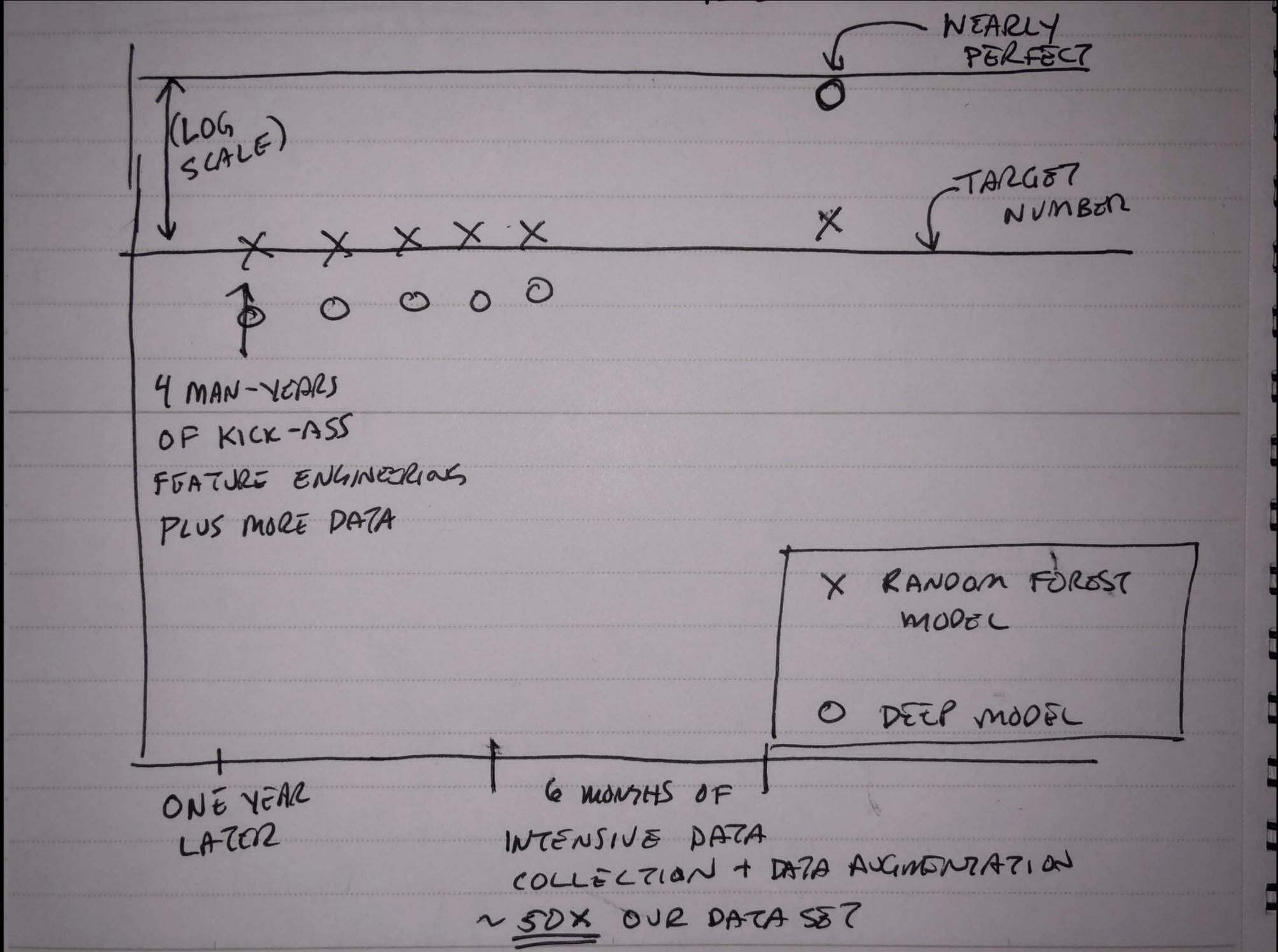
Hand Tracking Accuracy



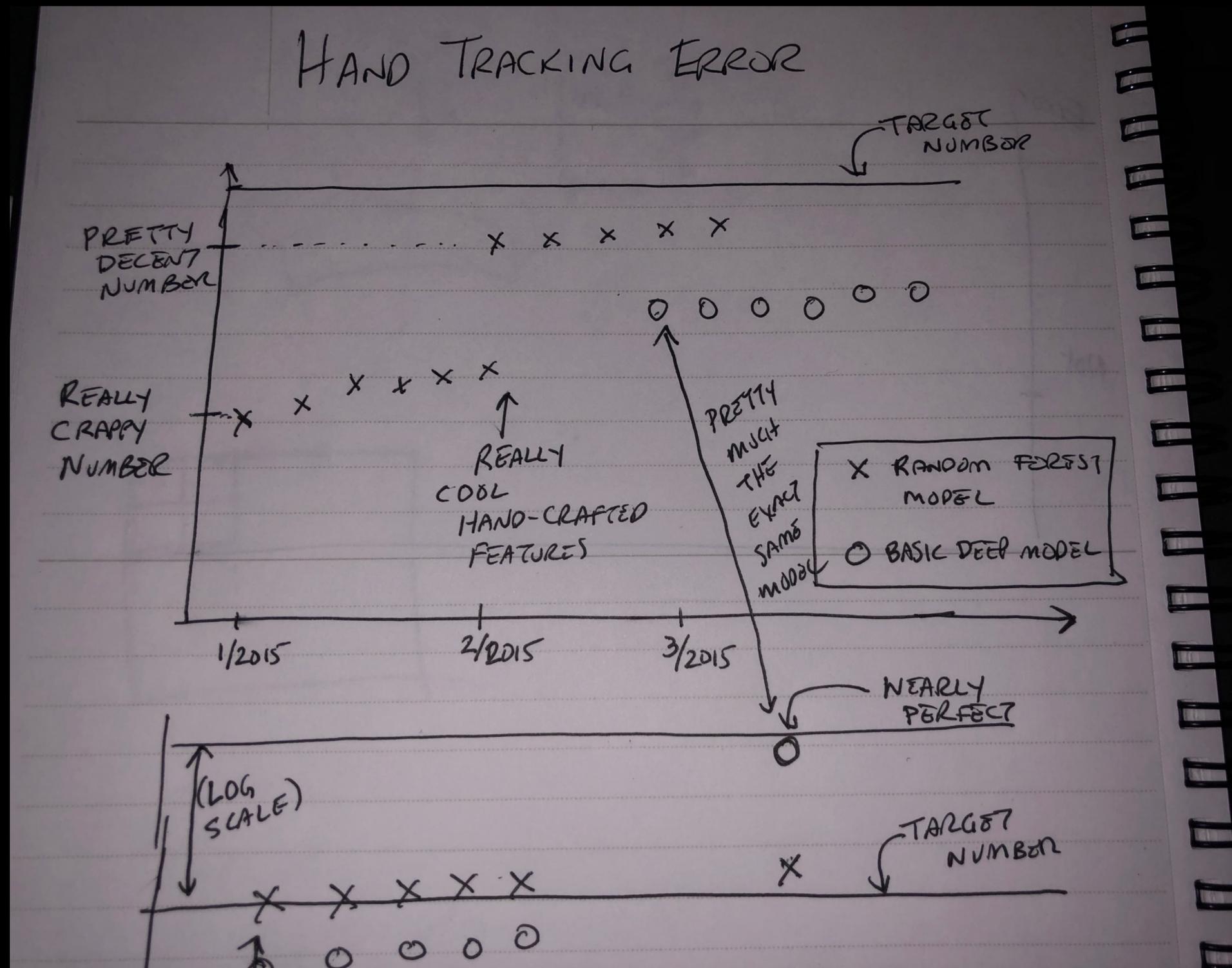
Case Study: Hand Tracking @ Magic Leap

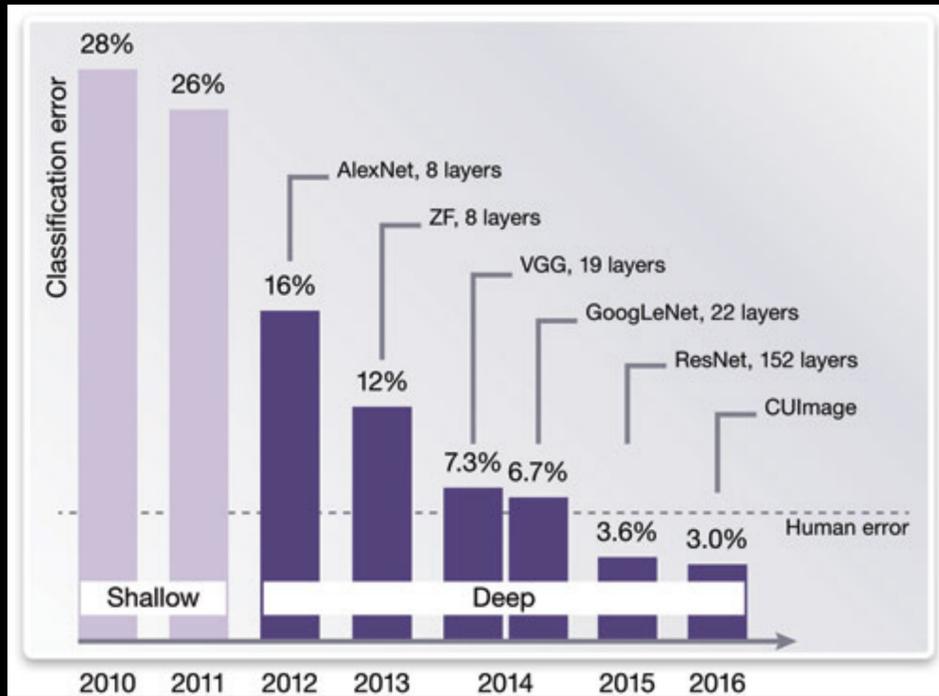


Case Study: Hand Tracking @ Magic Leap

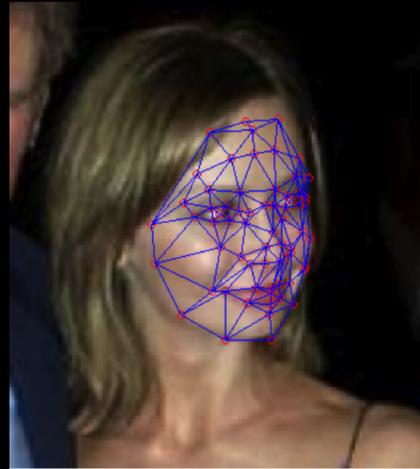


Case Study: Hand Tracking @ Magic Leap





ILSVRC Image Classification Error



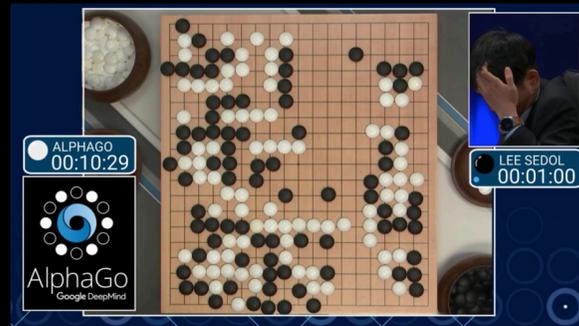
DeepFace 2014



OpenPose 2017



Deep Q Learning 2015



AlphaGo 2016

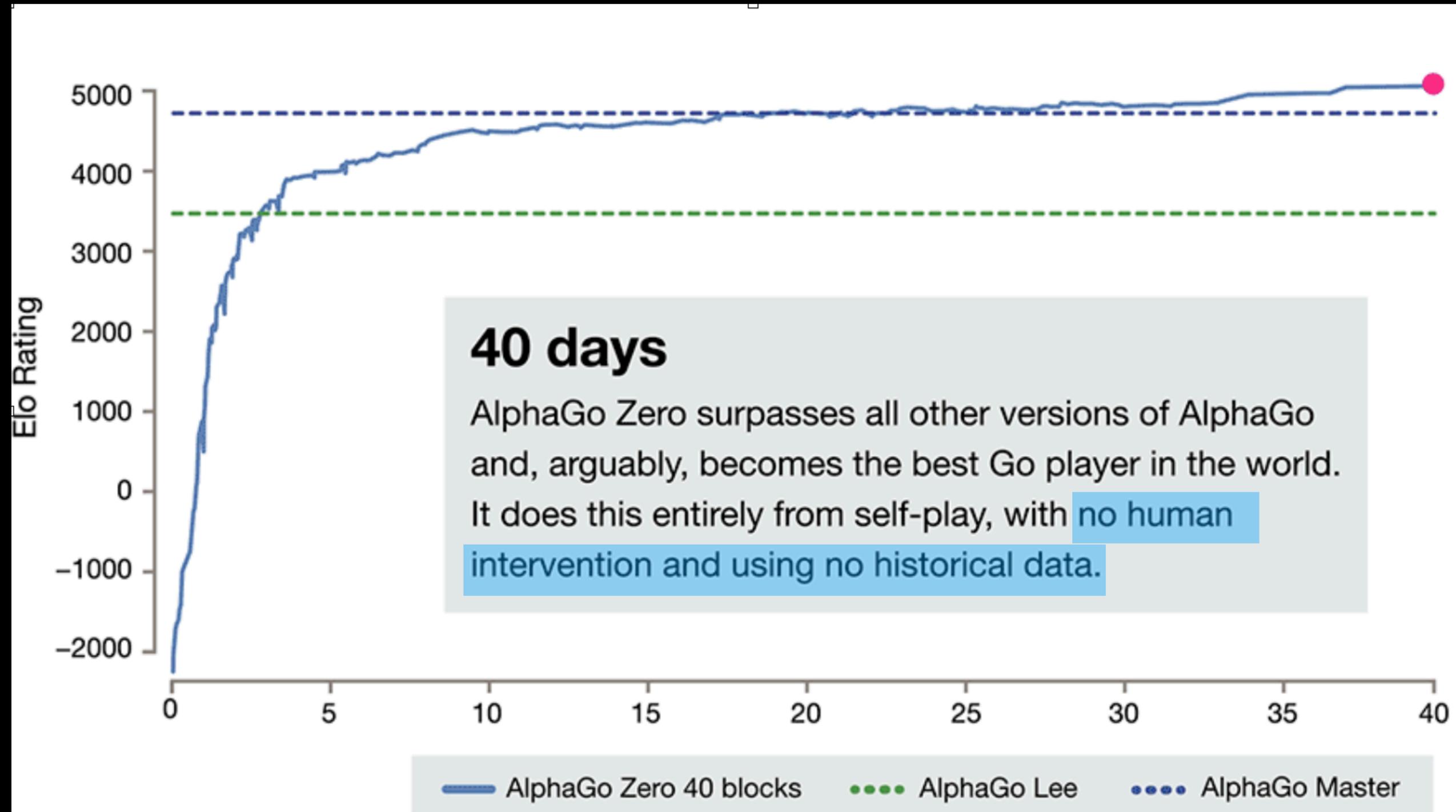


AlphaGo Zero 2017



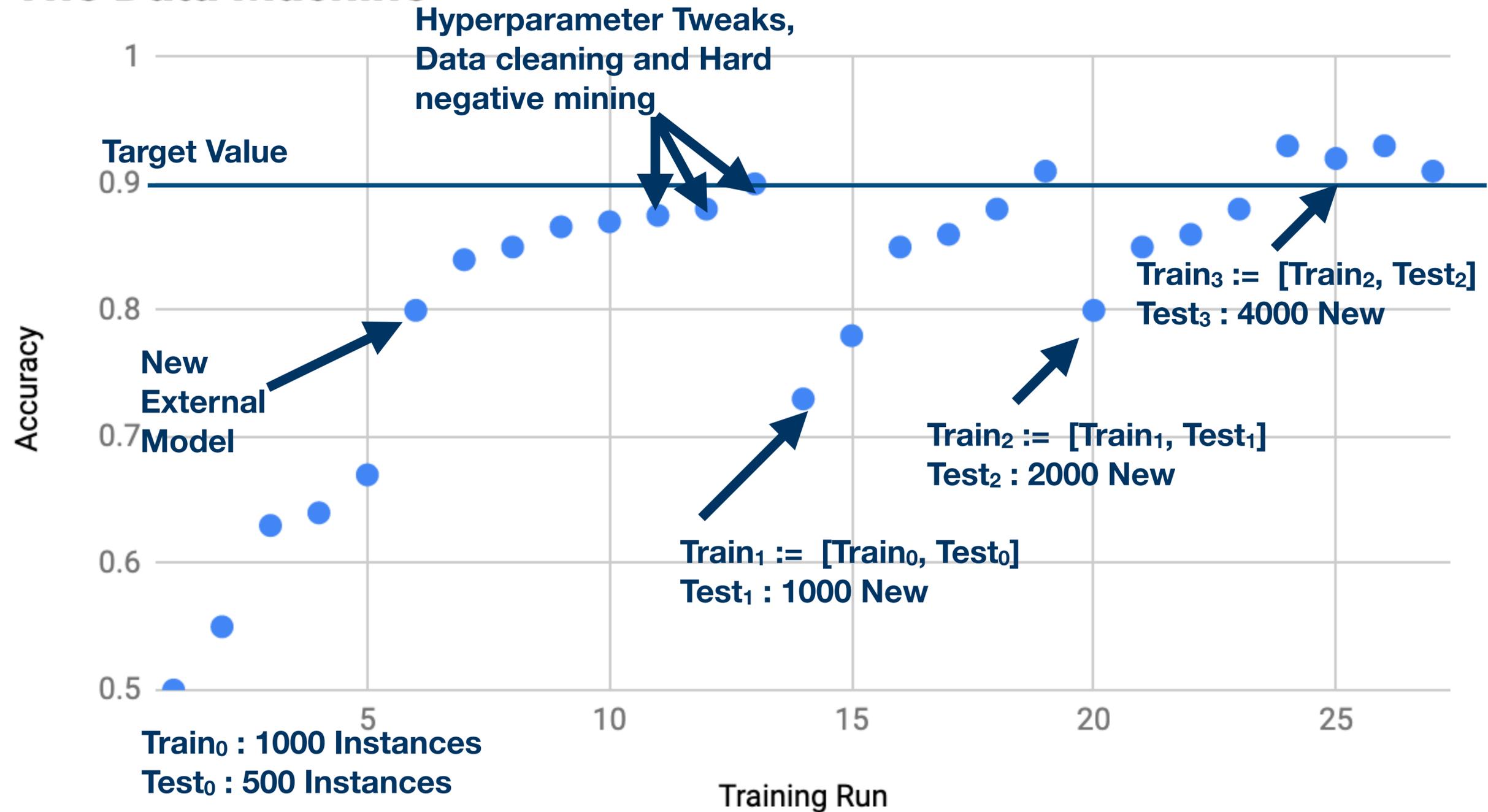
Deep Perception

Alpha Zero



How Do We Know When We Are Done?

The Data Machine



Summary

- **Avoid non-deep methods:** These complicate the system and therefore are error prone
- **Invest in a data machine:** Have the ability to collect new data at will, augment it, clean it, label it easily and cheaply
- **Stay up to date on recent research:** Follow leaders on Twitter, periodically pull down open code and test on your problem. Don't get too attached to your model.
- **Tweak hyperparameters:** But only while you are "twiddling your thumbs"
- **Detect when you are done:** Incrementally purge and double your test set until convergence