Designing Machine Learning Systems - A Book By Chip Huyen

Chapter Summaries and Key Learnings

This Book takes a Systems approach at the ML Engineering or MLOps processes. This means that all the components like Data Stack, Deployment, Model Monitoring, Model Maintenance, Infrastructure are being looked at holistically rather than just Model development.

Chapter - 1 : Overview of Machine Learning

- ML Algorithm is a small part of an ML System in Production
- Other Key parts include -
  - Business Requirement for the project
  - Interface with users and developers
  - Data availability and sourcs
- To Operationalize means to bring into Production and include developing, monitoring and maintaining the system.

When to use ML

- When there is data to **Learn** from
- There are **Patterns** in the Data. More complex patterns, better it is.
- Problems that require Predictive answers. Large quantity of cheap but approximate predictions. Compute intensive problems.
- Unseen data should share Patterns with the training data.
- If the task or Problem is Repetitive.
- When the cost of Wrong Predictions is Cheap like in Recommendation engines.
- When the benefit of correct predictions outweigh cost of wrong predictions.
- Data or Compute or Infrastructure or Predictions are at Scale.
- Constantly changing Patterns in the Problem space.

When Not to use ML

- It’s Unethical.
- Simpler solutions exist
- It’s not cost effective.

We can also try and break down our problem into smaller ones and use ML to solve them individually.

Some Use-cases of ML

- Recommender System
- Predictive Typing
- Photo Editing
- Phone Authenticating
- Machine Translation
- Smart Personal Assistants
- Smart Security Cameras

Enterprise ML use-cases

- Enterprise ML application use-cases can have stricter Accuracy requirements than Consumer ML applications
- Examples-
    - Fraud Detection
    - Price Optimization
    - Forecasting Customer Demand
    - Identifying Potential Customers
    - Churn Prediction
    - Sentiment Analysis
    - Medical applications like detecting skin cancer, diabetes.

ML in Research Vs. ML in Production: Different Requirements

- Research: Focuses on attaining state-of-the-art on Benchmark datasets.
  Production: Multiple stakeholders have multiple requirements
- Usually a project would involve ML Engineers, Sales team, Product Manager, Infrastructure Engineers, and Manager
- ML engineers need to understand expectations of different stakeholders and prioritise between them.
- Improvement in model performance at the cost of complexity needs to be looked from the perspective of corresponding improvement in sales revenue or user experience.
- More accurate models can come at a cost of compactness, fairness and energy efficiency.

ML in Research Vs. ML in Production: Computational Priorities

- Making mistake on focussing too much on Model dev rather than on model deployment and maintenance.
- Research: Prioritises Fast Training,
  Production: Prioritises Fast Inferences.
- Research: Throughput or number of queries / samples processed in one second matter more.
  Production: Latency or the duration that a query or a batch takes to return the results, matters a lot.

ML in Research Vs. ML in Production: Data

- Research: Dataset are often benchmarked, clean, formatted, static in nature, and open source scripts or processing pipelines are available.
  Production: Data is lot more noisy, unstructured, shifting in nature., and can be biased as well. Also User data comes with privacy and regulatory concerns.
- Research: Historical Data.
  Production: Data is constantly generated.

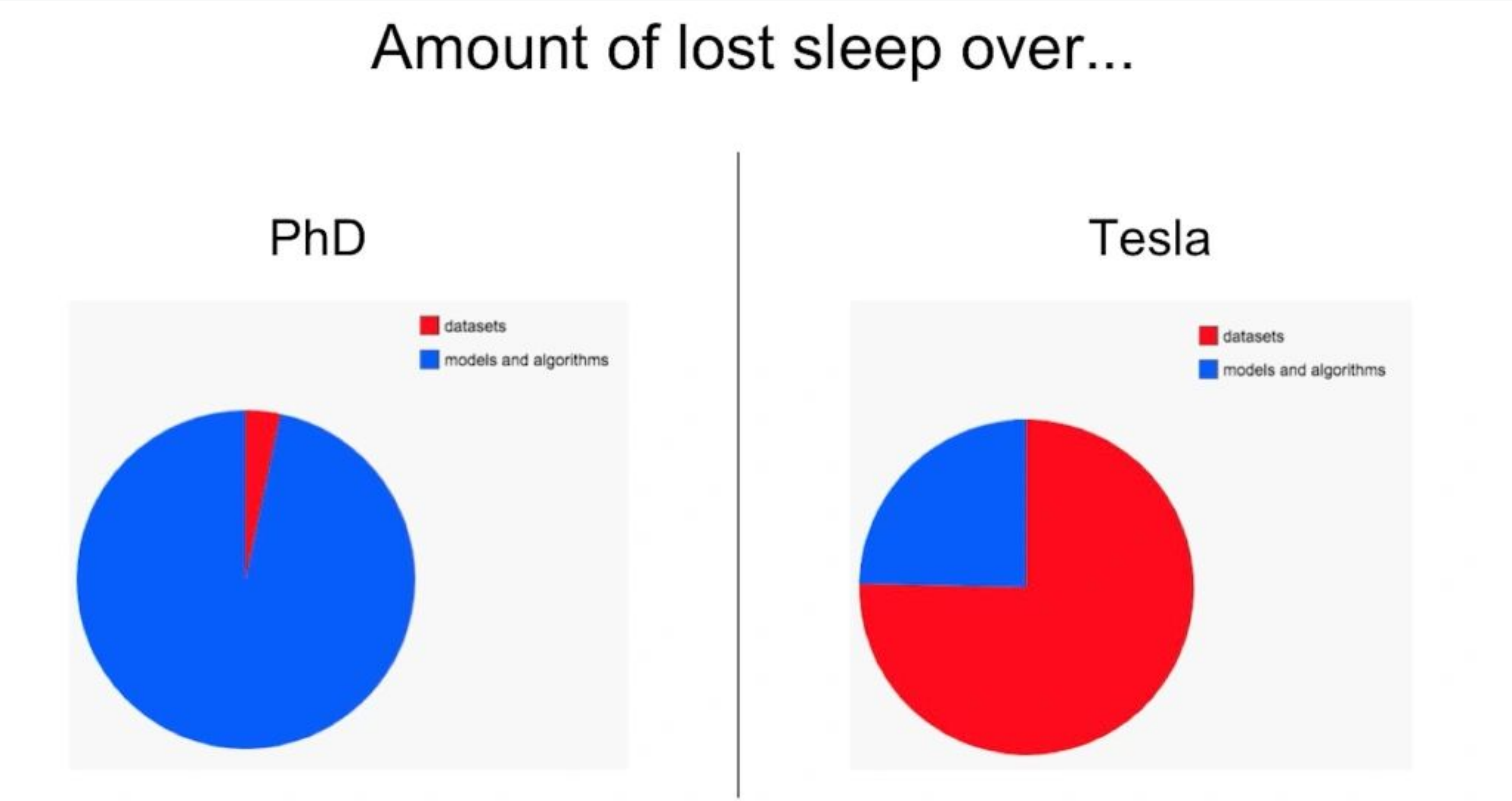

Image: Karpathy’s Sleep-lost Pie Chart
Credit: https://electrek.co/2018/06/11/tesla-ai-director-insights-autopilot-computer-vision-neural-net/

ML in Research Vs. ML in Production: Fairness

- Research: Fairness of model is like an afterthought. No state-of-art metrics for Fairness yet. Production:
- Examples of biases in ML in Production - Loan applications getting rejected, Resume application getting ranked lower, mortgage might get high rates, *etc.*
- ML systems capture biases in historical data while encoding the patterns, thus propagating biases even further in future decisions.
- ML systems work at scale so biases in decisions can have very high impact.

ML in Research Vs. ML in Production: Interpretability

- Research: Model performance is preferred over interpretability.
  Production: Interpretability is a crucial requirement.
- Interpretability is important for business leaders as well as end-users to understand why a decision was made.
- Interpretability is important for ML engineers for debugging and improving their models.

ML Systems vs Traditional Softwares

- Software Engineering: Code and Data are separated and modularised.
  ML Systems: Part Code, Part Data, and Part a mixture of these two.
- Software Engineering: Only test and version your Code.
  ML Systems: Test and Version both, Code and Data
- Companies want to improve and increase their data to improve their models.
- ML Applications need to be adaptive and agile as data changes every so often. So Faster development and deployment cycles are needed.
- Size of ML models is reaching billions of parameters, which need los of gigabytes of RAM to load ML models into memory.
- Getting massive models on edge devices, and getting them to run fast enough are key challenges.
- Model monitoring in Production is difficult as well.