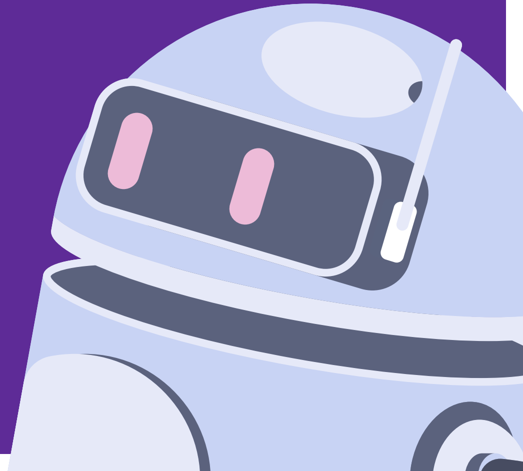
Debugging in the ML/AI Space

Strategies for Upcoming Talents

Shraddha Kulkarni
@shraddha-k27

Sunandan Barman
@sunandanbarman



Introductions

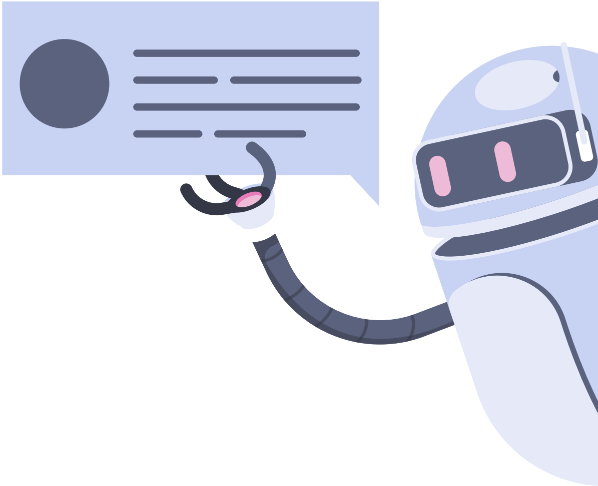
Shraddha Kulkarni

[@shraddha-k27]

Sunandan Barman

[@sunandanbarman]

Objectives



Importance of Debugging AI in Today’s Automation World

Why reliable and unbiased AI is crucial for business success

Why Debugging is a Must-Have Skill

How debugging makes you invaluable in the AI space

Bridging the Gap: Theory vs. Real-World ML in Big Tech

What textbooks don’t teach you about real-world AI challenges

Training New Engineers: Building Practical Debugging Skills

Strategies and resources for training engineers to tackle complex AI issues

Debugging AI in Today's Automation World

AI is Everywhere

From social media to self-driving cars, AI impacts millions. Debugging ensures these systems perform reliably and accurately.

Models Aren't Perfect

Data can be messy, leading to incorrect predictions. Debugging helps clean up errors and biases in models.

Optimizes Efficiency

Well-debugged systems are faster and more resource-efficient, crucial for large-scale operations like Amazon or Google.

Why Debugging is a Must-Have Skill

Prevents Big Mistakes

Automated systems make decisions without humans. Debugging minimizes costly errors (e.g., fraud detection in finance).

Continuous Improvement

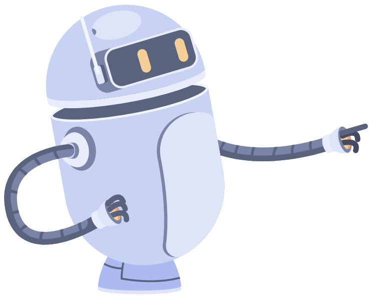
Models need constant updating. Debugging ensures they evolve without breaking down.

In-Demand Skill

Debugging AI/ML systems is a rare, high-demand expertise, opening career opportunities in the growing automation industry.

Bridging the Gap:
Theory vs. Real-World ML

Crucial for engineers transitioning into industry roles



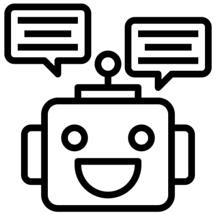
Look at these numbers...

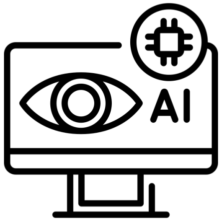
Over **277 million daily active users** worldwide

500 million daily active users, with 50 million requests per hour during peak times

2 billion monthly users, with 500 million daily users engaging heavily

Heavy-Duty ML Models Behind the Scenes





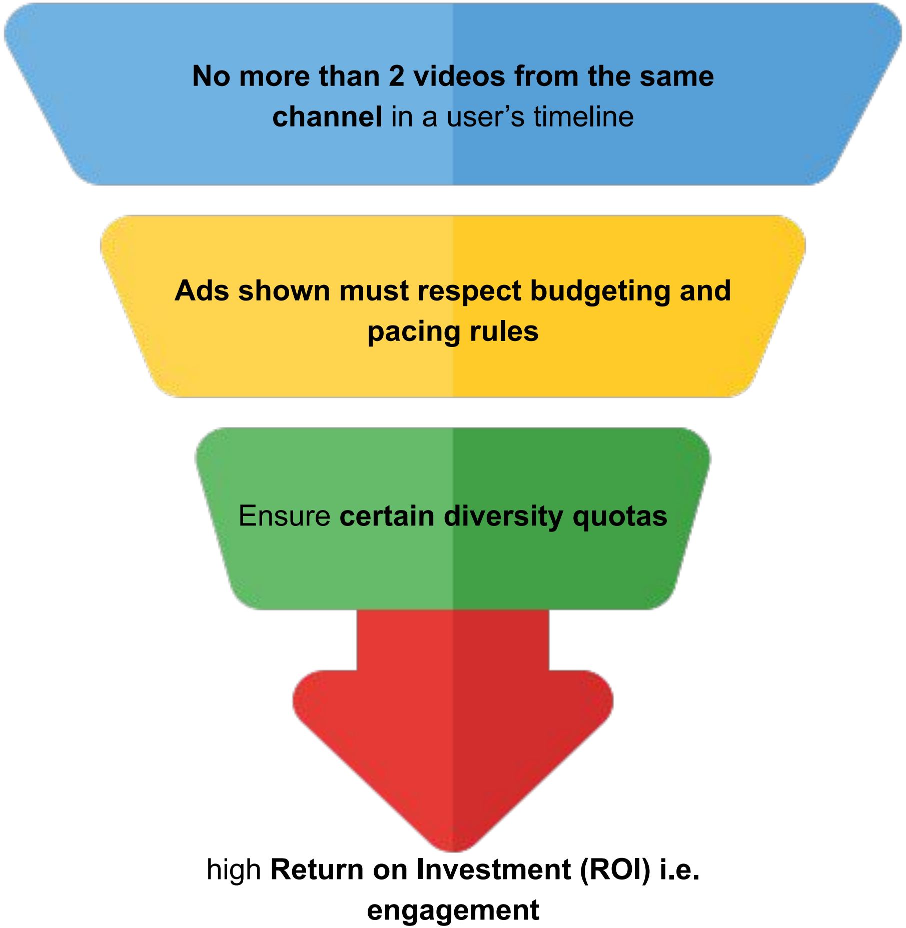
Model	Parameters	Use case
BERT	**110 million**	Bidirectional Encoder Representations from Transformers
GPT-3	**175 billion**	language model by OpenAI
T5	**11 billion**	Text-to-Text Transfer Transformer
ResNet-152	**60 million**	Residual Networks used in Image Classification and Object detection
YOLOv4	**64 million**	You Only Look Once - Real time object detection in video streams and image-based applications

The Data Pipeline: Privacy and Sensitivity Filtering

Privacy Law	Region	Details
GDPR	**EU**	Strict user consent rules, heavy fines (€20M or 4% of revenue), and robust rights for data access and deletion
CCPA & CPRA	**California, USA**	Grants rights to access, delete, and opt-out of data sales for California residents. Enhanced by CPRA with stricter rules
Bill C-27	**Canada**	Proposed update to strengthen PIPEDA, enhancing data access and privacy rights for Canadians
LGPD	**Brazil**	Influenced by GDPR, mandates consent and transparency, with fines up to 2% of revenue (capped at R$50 million)
Australia’s Privacy Act Reform	**Australia**	Under review, aiming to align closely with GDPR with stronger data rights and compliance

From Raw Data to Ranking

Business Rules in Action: A Real-World Example from YouTube



Continuous Monitoring and Real-World Validation

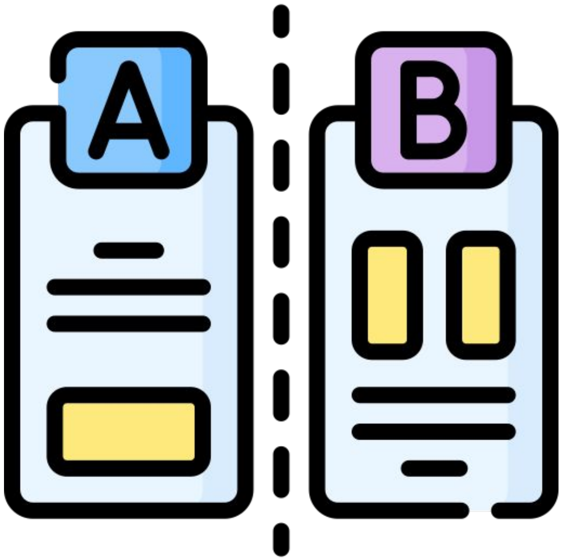
Unlike in school, where you measure accuracy using a simple test dataset, **real-time performance validation is impossible.**

- On YouTube, you **can't instantly validate** whether showing a certain set of ads led to better engagement or are biased.
- Often, the true impact is confirmed **only after weeks or even months** of monitoring performance and revenue metrics.

Example: Bias in Early ChatGPT Versions

In large-scale models like early versions of ChatGPT, **bias wasn't detected until real users started interacting with the system**. For example, some versions were found to lean towards certain political ideologies.

Model Deployment: Beyond Just Training



Before deploying a new model, companies run A/B tests on a small subset of users.

You need to prove to stakeholders that your model will:

- **Increase engagement metrics**
- **Improve prediction performance**
- **Not cause any disruptions**

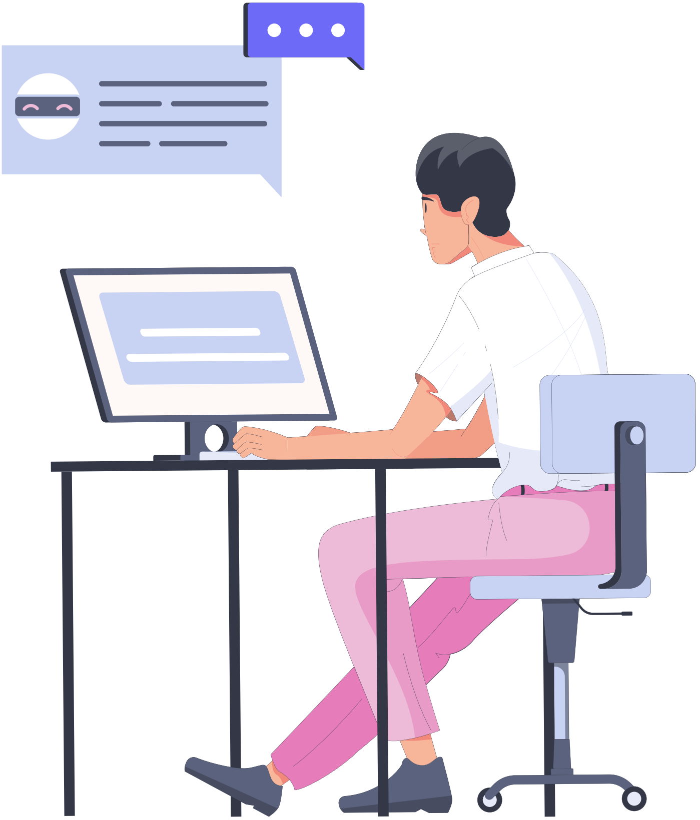
Real-World ML: More Than Just Accuracy

In **<u>theory</u>**, the goal is to **maximize accuracy** or **minimize loss**.

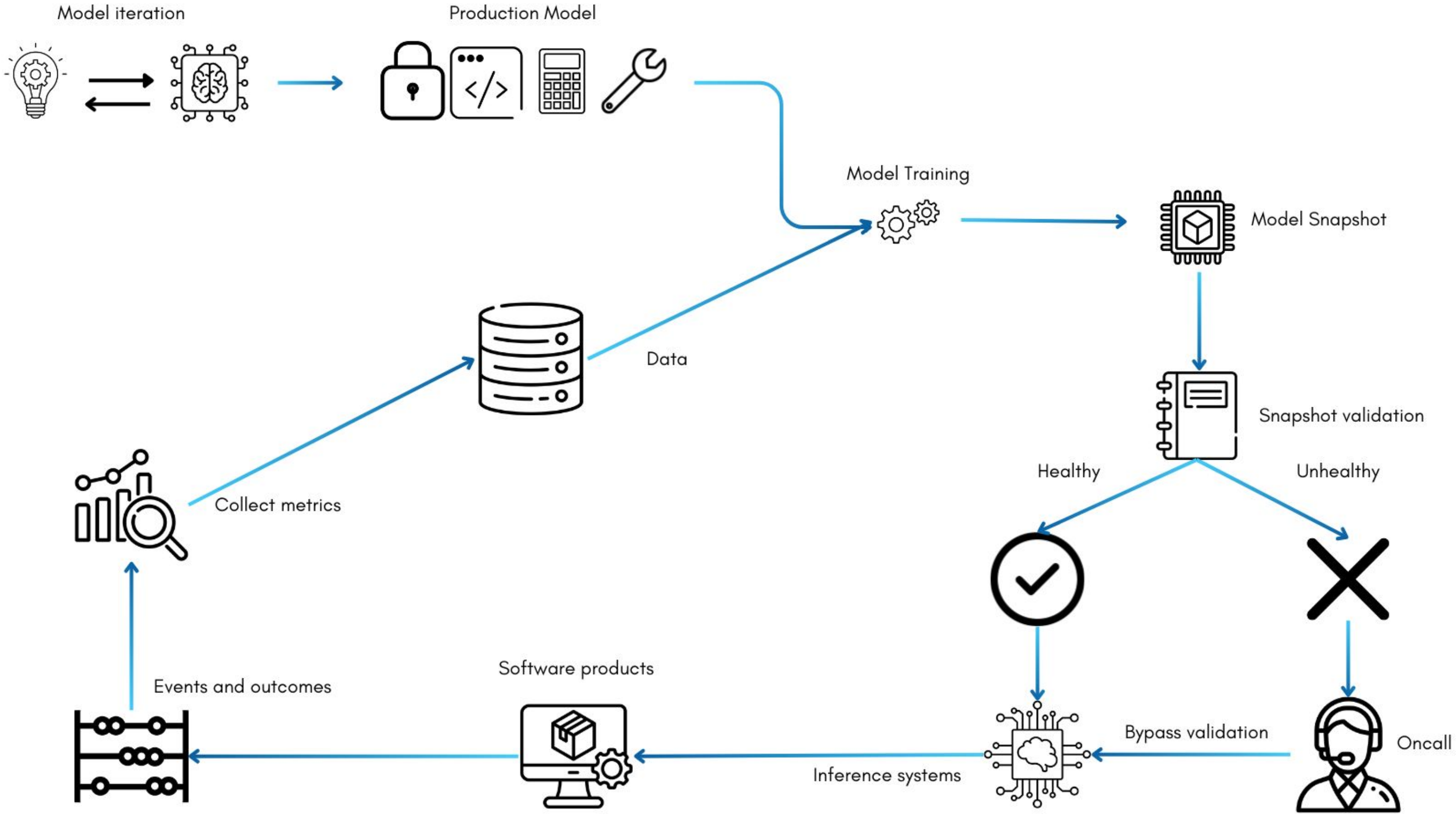
But in **<u>production</u>**, you need to consider system efficiency, privacy, real-world constraints, and business metrics. This is why real-world ML is about **making trade-offs** and ensuring the model **delivers value**, not just high accuracy.

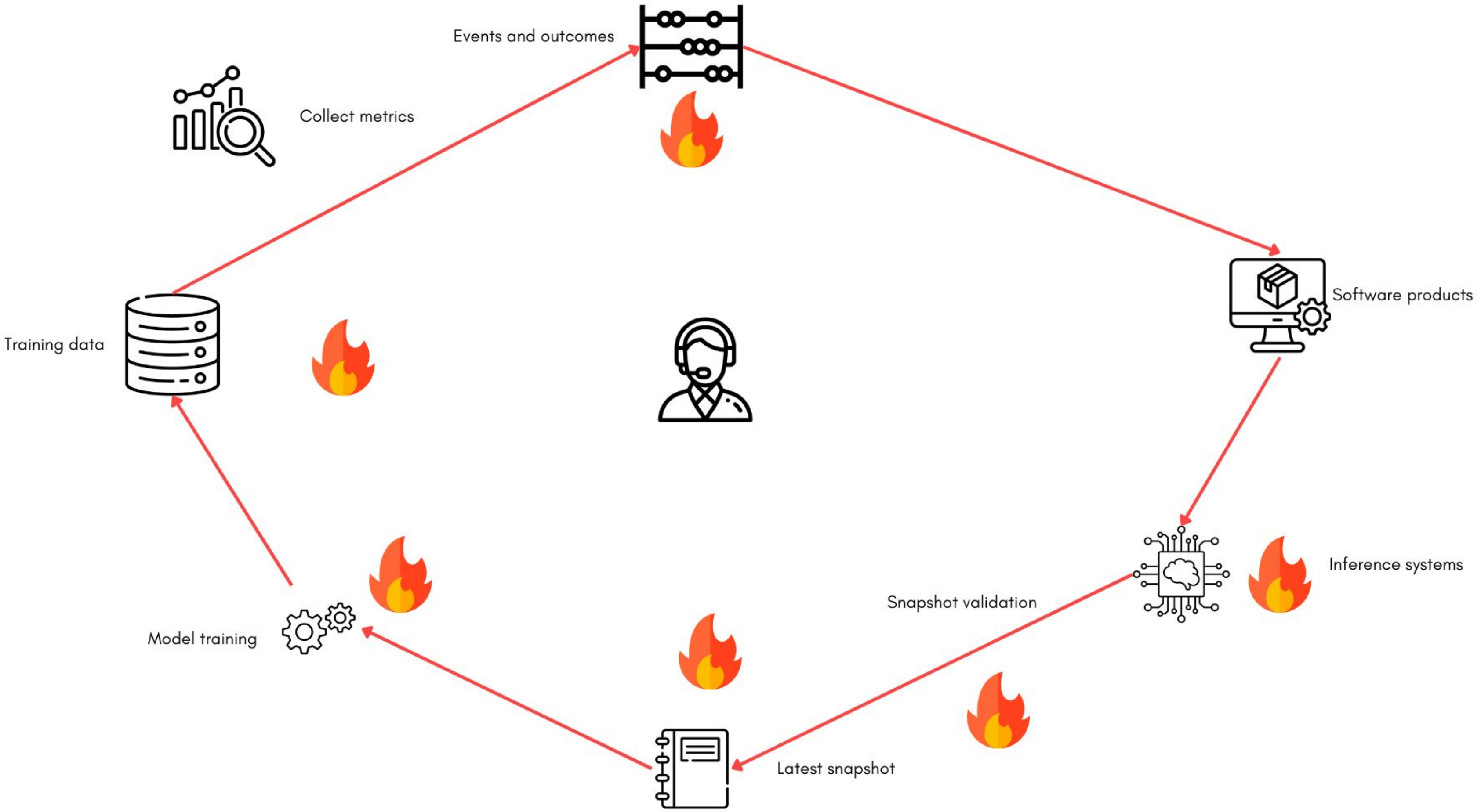
Training Junior Engineers to Debug AI / ML Systems

Building Practical Skills for the Real World

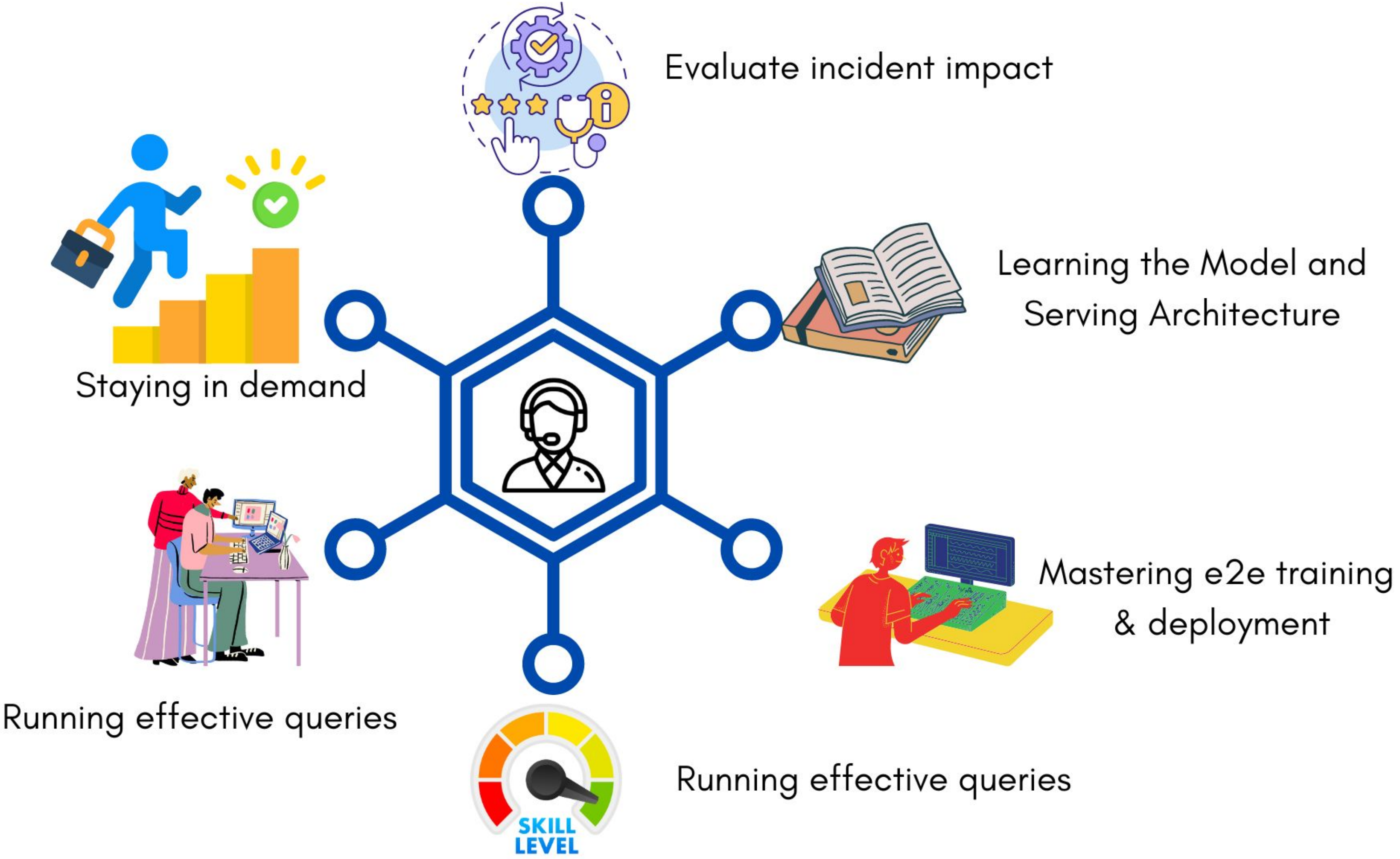


Machine learning systems in production

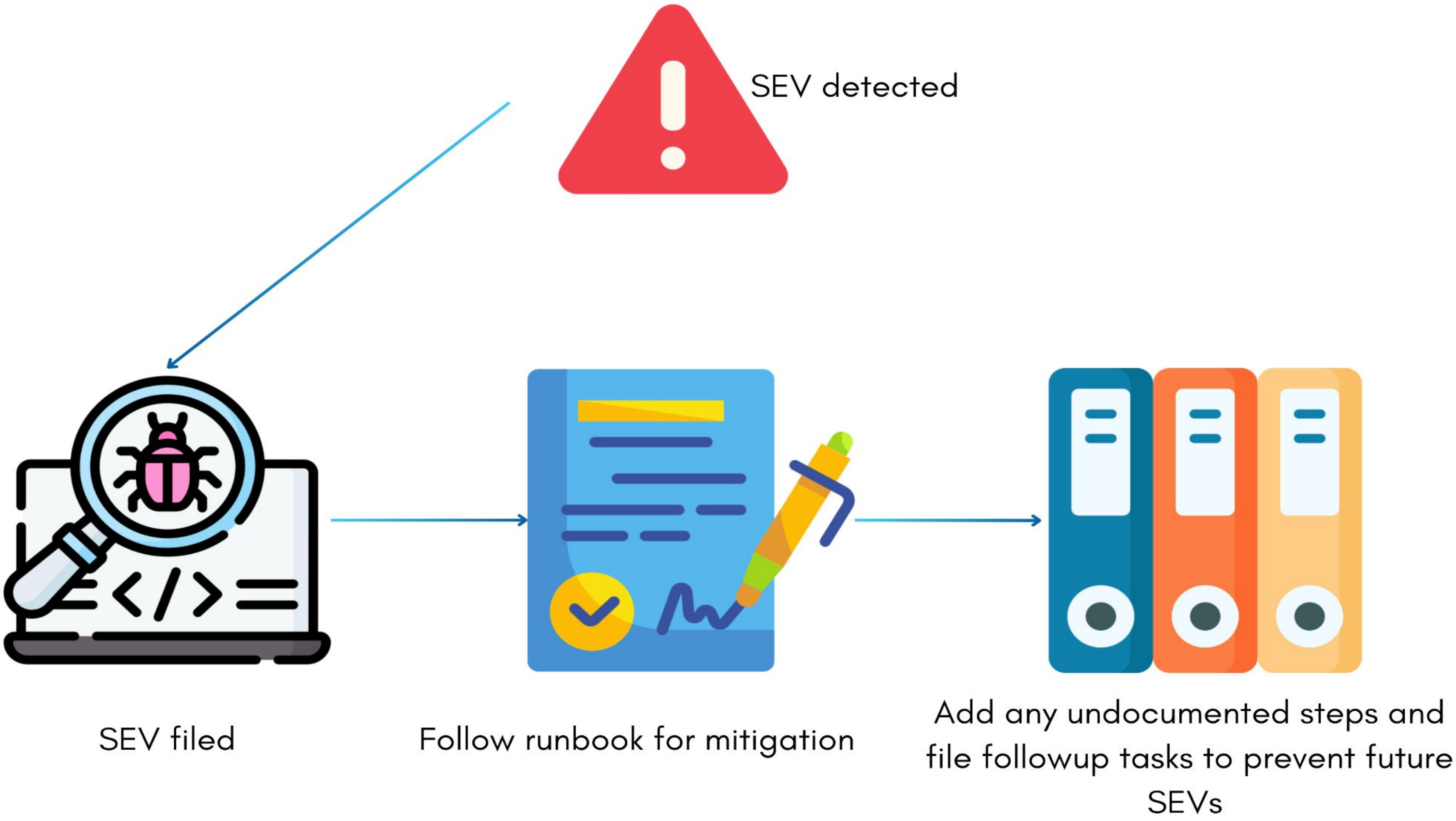




Events and outcomes
Collect metrics
Software products
Training data
Inference systems
Snapshot validation
Model training
Latest snapshot


Evaluate incident impact
Learning the Model and Serving Architecture
Staying in demand
Mastering e2e training & deployment
Running effective queries
Running effective queries
SKILL LEVEL

Evaluate incident impact & filing incident reports



Learning the Model and Serving Architecture

What They Need to Know:

- Understand components of AI system architecture, like model servers, data pipelines, and real-time inference.

Practical Training:

- Use architectural diagrams to show where common failures occur (e.g., latency spikes, data pipeline failures).

Example:

- Amazon's AI hiring tool favored male candidates due to a bias introduced in its training data pipeline, which wasn't identified early in the architecture review (Scalable Path).
- Walk through exercises diagnosing issues like data latency between feature stores and serving layers.

Tip:

- Walk through exercises diagnosing issues like data latency between feature stores and serving layers.

Resources:

- **Google's ML System Design Course**
- **AWS Machine Learning Architecture Blog**

Mastering the End-to-End (E2E) Training and Deployment Flow

- **Key Concepts**:
    - Teach them about data preprocessing, model training, evaluation, deployment, and continuous monitoring.
- **Why It Matters**:
    - Understanding the E2E flow is critical for identifying root causes—e.g., when the Google Photos model labeled Black individuals as "gorillas," the problem originated in the training data and model evaluation stages (Arize AI)
- **Example**:
    - A junior engineer retraining a chatbot model must understand where new data is coming from, how the model is updated, and what downstream services are affected.
- **Tip**:
    - Create small projects where juniors go through this flow on a simpler dataset before tackling complex models.
- **Resources**:
    - **"Building Machine Learning Pipelines" by Hannes Hapke**
    - **Databricks Engineering Blog** for deployment workflows.

Building Debugging Skills — Running Effective Queries

- **Why Queries Matter**:
  - AI can generate code, but knowing **what to query for** is a human skill.
- **Examples of Queries**:
  - **Bias Analysis**: Check if a model's predictions disproportionately favor a group, like Google Translate's gender bias, which defaulted to male pronouns for doctors and female pronouns for nurses (Scalable Path)
  - **Latency Bottlenecks**: Query to identify slow API endpoints or missing features in real-time models.
- **Real-World Example**:
  - If a recommendation system suggests irrelevant items, run a query to check if recent user preferences are missing or misaligned.
- **Tip**:
  - Encourage them to practice on real datasets and build intuition for critical data points.
- **Resources**:
  - **Mode Analytics SQL Tutorials**
  - **"SQL for Data Analysis" by Cathy Tanimura**

Finding a Mentor and Building a Support Network

- **Why Mentors Matter**:
  - Learning to debug complex systems can be overwhelming. A good mentor can guide, share best practices, and help juniors avoid common pitfalls.
- **How to Find a Mentor**:
  - Look within the organization or connect via LinkedIn and engineering communities (e.g., StackOverflow).
- **Example**:
  - Encourage shadowing sessions or attending debug war rooms, where they can learn from incidents like Amazon's AI tool bias issue, which required multiple teams and stakeholders to diagnose (Scalable Path)
- **Tip**:
  - Suggest joining AI/ML communities like **KDnuggets** or **AI Breakfast Club** for peer support.
- **Resources**:
  - **Women in Machine Learning & Data Science (WiMLDS)**
  - **KDnuggets Community Forums**

Staying in High Demand in an AI-Heavy Market

- **Key Skills to Focus On**:
  - Debugging expertise, understanding system design, and strong coding/querying skills.
- **Develop a Niche**:
  - Specialize in areas like **MLOps**, **Data Engineering**, or **Model Interpretability**.
- **Continuous Learning**:
  - AI tools are evolving rapidly. Keep up with trends and learn new frameworks.
- **Real-World Impact**:
  - Understand how models like the COMPAS system affected real-world outcomes and how fixing such issues can significantly improve a company's reputation and avoid legal pitfalls (Scalable Path).
- **Resources**:
  - **Books**: *"Designing Machine Learning Systems"* by Chip Huyen, *"The Hundred-Page Machine Learning Book"* by Andriy Burkov.
  - **Podcasts**: *Data Skeptic*, *Lex Fridman Podcast*.
  - **YouTube Channels**: *StatQuest with Josh Starmer*, *Two Minute Papers*.
  - **Engineering Blogs**: Google AI Blog, OpenAI Blog, Distill.pub.

Great culture

Great Team

Great tooling

Conclusion

CONF42