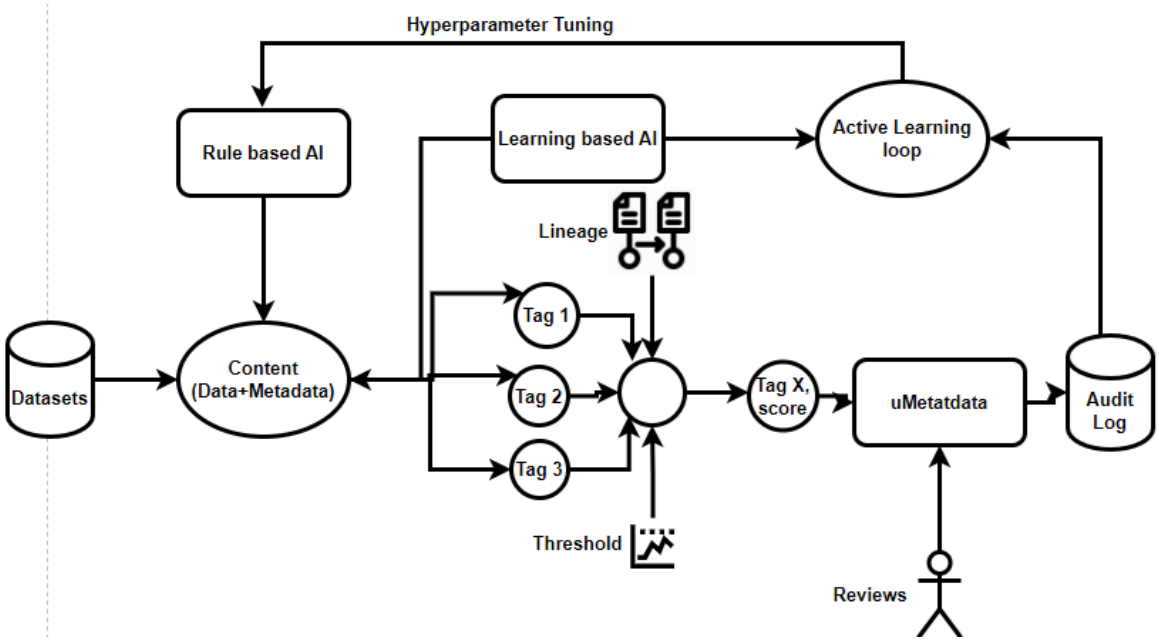
Data categorization is essential for privacy and security, enabling effective access control, encryption, and data lifecycle management. However, the sheer scale of modern datasets, ongoing data generation, and the complexity of manual tagging make traditional approaches impractical. Engaging data owners for classification is challenging due to the volume of tags and the risk of miscategorization. To address these challenges, Uber developed **DataK9**, an AI/ML-powered auto-categorization platform that minimizes human involvement. This solution enhances efficiency, reduces costs, and ensures scalable data classification across Uber's vast data ecosystem.

Challenges in Data Categorization and Uber's AI/ML-Based Approach

- Column Names Aid Categorization → While numeric data alone lacks inherent meaning, column names (e.g., *Latitude, Longitude*) provide valuable context for classification.
- Generic Column Names Create Ambiguity → Columns with vague names (*e.g., Note*) could represent multiple data types, such as cost, balance, or distances, making classification difficult.
- Precision-Based Categorization → Location data with high precision (three or more decimal places) may be classified as *Highly Restricted* if linked to personal identity.
- Contextual Challenges in Address Data → Without explicit context, distinguishing between *personal and business addresses* remains difficult, even though both are categorized as addresses.
- AI/ML-Based Probabilistic Approach → Due to these complexities, Uber employs AI/ML-driven probabilistic methods to improve data categorization rather than relying on rigid rule-based classification.



Strategy

The process follows a hybrid approach, combining manual expert validation with automated tagging, structured into three key stages:

1. Manual Labeling of Golden Datasets → A small subset (<1%) of data is categorized manually by domain experts to serve as high-quality training data.

2. Model Training & Learning → The AI/ML models use these golden datasets to learn patterns, associations, and context for categorization.

3. Automated Tagging at Scale → The trained model then auto-tags the remaining (>99%) data, significantly reducing manual effort while maintaining accuracy.

Building a Baseline for AI-Driven Data Categorization

To ensure high categorization accuracy, Uber employs a baseline approach by manually tagging a small subset (<1%) of critical datasets (~1,000 tables). These "golden datasets" serve as a foundation for measuring automation accuracy and offer key benefits:

1. Model Training → The manually labeled datasets form the training baseline for AI/ML models, enabling them to learn accurate categorization patterns.

2. Risk Mitigation → Business-critical datasets are manually tagged by domain experts to prevent misclassification and reduce operational risks.

3. Avoiding Statistical Bias → The selection of golden datasets is done randomly to prevent biases that could skew model performance and lead to systemic classification errors.

Training Process for AI-Driven Auto-Categorization

For the remaining 99% of datasets (>400K), Uber leverages DataK9, an automated categorization system. The training process consists of two key sub-stages:

1. Iterative Model Training & Evaluation

- The AI model is trained on a subset of golden datasets and tested against the remaining golden datasets.
- Performance is evaluated using accuracy, precision, recall, F2-score, etc.
- If metrics are unsatisfactory, model tuning and re-evaluation continue until optimal accuracy is achieved.

2. Rule-Based Detection & Auto-Adjustment

- Domain experts define tagging rules and iteratively test & refine them.
- Initially, human oversight ensures rule accuracy, but over time, the rule adjustment process is automated for better predictions.

Deploying DataK9 in Production

Once DataK9 achieves >90% accuracy and >85% F2-score, it is ready for mass data categorization. However, defensive measures are taken in the early deployment phase to ensure accuracy and minimize risks:

1. Human Review for Early Auto-Categorization → Automatic ticket creation assigns newly categorized datasets for manual review by data owners or privacy experts before finalization.

2. Continuous Monitoring & Misclassification Analysis → The system tracks misclassification rates, ensuring errors are identified and corrected before scaling further.

3. Gradual Scaling & Mass Deployment → If review feedback confirms satisfactory performance, the auto-categorization system is rolled out to hundreds of thousands of datasets for full deployment.

Architecture

AI-Based DataK9 Architecture for Automated Data Tagging

The DataK9 framework leverages AI/ML techniques across multiple stages to tag datasets efficiently. The key components of the architecture include:

1. Feature Extraction → Essential features of a dataset (e.g., column names, data patterns, metadata) are identified and gathered for analysis.

2. ML/AI-Based Tag Scoring → The system applies machine learning algorithms to calculate weights/scores for assigning a column to a specific category.

3. Multi-Signal Decision Making → Multiple signals (e.g., statistical patterns, metadata context, and rule-based criteria) are combined to reach the final tagging decision.

4. Active Learning & Continuous Model Tuning → The system analyzes misclassifications, adjusts categorization rules, and re-trains the ML model to improve tagging accuracy over time**.**

Key Features of DataK9 for AI-Driven Data Categorization

DataK9 leverages multiple data attributes as features in its ML/AI classification techniques to enhance accuracy and scalability.

1. Metadata Analysis → Extracts field names & data types (e.g., "request_latitude" (double), "email" (string)) to infer category relevance.

2. Data Sampling & Content-Based Classification → Analyzes ~1% of dataset records per scan, using cell values to determine column categories.

3. Context Awareness → Evaluates record-level relationships, table names, and database structure to provide additional signals for classification.

4. Data Lineage Tracking

- Table-Level Lineage → Identifies source tables that contributed to derived tables (e.g., "uber_rides" derived from "drivers" & "riders").
- Column-Level Lineage → Maps column dependencies across tables to improve tag predictions, integrating with data platforms like Hive & Spark.

Matching Strategy in DataK9 for AI-Driven Data Categorization

DataK9 continuously crawls, scans, and analyzes data elements across storage systems, using various matching techniques to assign tags accurately.

1. Individual vs. Aggregate Decision

- Cell-Level Analysis → Examines individual data points (e.g., john@gmail.com detected as an email).
- Column-Level Decision → Aggregates individual matches to determine the overall column category (e.g., if 80% of values match emails, the column is tagged as PII).
- Avoiding False Positives → Ensures context-aware classification to prevent misclassification (e.g., an email in a "promotion_code" column may not be PII).

2. Probabilistic Decision Making

- Assigns a confidence score to each potential match.
- The higher the score, the stronger the tag association.

3. Negative Scoring for Mismatches

- Data elements that are unlikely to belong to a tag are given a negative score (e.g., tables under the "products" database are unlikely to contain PII).

Rule-Based AI in DataK9 for Auto-Categorization

DataK9 applies rule-based AI alongside machine learning to categorize datasets. The rule engine operates using hand-crafted rules, designed with 40+ different tags and fundamental data features. The key components include:

1. Matching Techniques for Rule-Based Classification

- Bloom Filter Match → Checks if a data value exists in a predefined probabilistic filter (e.g., matching addresses used in Uber deliveries).
- Dictionary Match → Looks up predefined dictionaries (e.g., airport codes, city names) to determine category relevance.
- Pattern Match → Uses regular expressions to identify structured data (e.g., detecting emails via regex).

2. Context-Based Matching

- Record-Level Context → Looks at relationships within the same record (e.g., latitude alone is non-sensitive, but with longitude, it's precise location data).
- Table-Level Context → Uses table names to infer sensitivity (e.g., "full name" in "rider" table is sensitive, but in "virtual_machine" table, it is not).
- Database-Level Context → Identifies database-level patterns (e.g., a database named "finance" likely contains transaction data).

3. Data Type Matching

- Enforces data type constraints (e.g., an "email" column must be of type string; numeric columns are skipped**).**

Rule Language for Defining Data Categorization in DataK9

DataK9 supports a standardized rule definition language for rule-based AI classification. A rule consists of three types of sub-rules, each assigned a matching score by domain experts:

1. Column Value Matching (columnValueMatch) → Examines individual data values with the following techniques:

- Bloom Filter Match → Checks if a value exists in a precomputed Bloom filter.
- Dictionary Match → Compares values against a predefined list (e.g., common city names).
- Length Range (lengthRange) → Filters values outside a defined length (e.g., latitude must be 3-10 characters).
- Value Range (valueRange) → Excludes values outside a valid range (e.g., age must be between 0-125).
- Pattern Match → Uses regular expressions to detect structured formats (e.g., email regex pattern).

2. Metadata-Based Column Name Matching (columnNameMatch)

- Evaluates column name patterns and data types to confirm the categorization.
- Can exclude certain column names/types to prevent false positives.

3. Context-Based Matching (contextMatch)

- Resource Context (resourceContext) → Identifies database & table patterns (e.g., a database named “finance” suggests transaction data).
- Category Context (categoryContext) → Ensures dependent relationships exist (e.g., Level 1 location data must have personal identifiers in the same dataset).

Implementation & Storage

- Rules are stored in YAML files for easy configuration.
- Alternatively, rules can be stored in a database for dynamic updates.

Learning-Based AI in DataK9 for Automated Data Categorization

DataK9 applies supervised learning for multi-label classification, leveraging metadata and cell content to predict data tags.

Key Steps in Learning-Based AI

1. Training the Model

- Uses labeled datasets to train ML models separately on metadata and data values.
- Incorporates feature engineering & input transformations for better learning.

2. Model Evaluation & Refinement

- Assesses performance using accuracy, precision, recall, and F2-score.

- If metrics are unsatisfactory, retrains the model by tuning hyperparameters or modifying the algorithm.
- Thresholding techniques fine-tune precision-recall trade-offs.

3. Handling Low-Support Categories

- Enhances models with additional manual labeling for underrepresented categories.
- Combines similar categories when manual labeling does not improve accuracy.

4. Final Sanity Check Before Deployment

- Runs train/test experiments to validate model performance.
- Ensures the model meets established accuracy thresholds before deployment**.**

ML Algorithms & Initial Experimentation

Tested Algorithms: Linear Support Vector Machine (SVM), K-Nearest Neighbors (KNN), Naïve Bayes.
Best Performer: Linear SVM (but requires further tuning for low-support categories**).**

Aggregating Signals in DataK9 for Accurate Data Categorization

DataK9 generates multiple signals for each column, assigning tags with scores using various AI techniques. The final classification is determined through weighted aggregation of these signals.

Sources of Tagging Signals

1. Rule-Based AI → Generates tags & scores based on predefined rules.
2. Learning-Based AI → Predicts tag-score pairs using metadata & column values.
3. Lineage Service → Provides additional insights based on data origin & dependencies**.**

Weighted Aggregation Process

1. All scores per tag, per column are aggregated using a weighted method.
2. Weights are empirically determined and parameterized in the rule definition.
3. Final tag assignment is based on comparing the aggregated score to a thresho**ld.**

Learning-Based Feedback Loop in DataK9

DataK9 continuously improves its ML/AI-based categorization by implementing an auto-learning feedback loop to correct errors and enhance accuracy.

Steps in the Feedback Loop

1. Mistake Identification

- After auto-tagging, data owners or privacy experts can modify incorrect labels using a UI.
- All modifications are logged in an audit trail for future analysis.

2. Pattern Recognition & Adjustments

- Identifies common mistake patterns to refine AI/ML predictions.
- Adjusts rule database parameters based on error trends.
- Fine-tunes model training parameters to improve classification accuracy.

3. Transition to Full Automation

- Starts with human-in-the-loop supervision for adjustments.
- Gradually automates rule & model refinements based on real-world feedbac**k.**

Aggregating Signals in DataK9 for Accurate Data Categorization

DataK9 generates multiple signals for each column, assigning tags with scores using various AI techniques. The final classification is determined through weighted aggregation of these signals.

Sources of Tagging Signals

1. Rule-Based AI → Generates tags & scores based on predefined rules.
2. Learning-Based AI → Predicts tag-score pairs using metadata & column values.
3. Lineage Service → Provides additional insights based on data origin & dependencies.

Weighted Aggregation Process

1. All scores per tag, per column are aggregated using a weighted method.
2. Weights are empirically determined and parameterized in the rule definition.
3. Final tag assignment is based on comparing the aggregated score to a threshold.

Measuring the Accuracy

In the realm of auto-categorization, accuracy reigns supreme, and thus, we've made it our priority to meticulously measure and disclose various levels of metrics tailored to our audience's needs. At the pinnacle of our reporting hierarchy, we shall unveil the following key metrics:

Standard Metrics for Classification: The quality of our automation frameworks is critical to evaluating and tracking the progress. However, it is impossible to measure the quality of auto-categorization without a proper baseline. Hence, we also proposed that manual categorization of <1% of datasets covering all data tags by owners/experts would provide a realistic baseline. In the ML/AI world, classification is well-studied literature, and the experts defined a set of metrics to gauge the quality of a classification method. We show them in Figure 8 with a confusion matrix and describe them below:

1. Accuracy: Classification accuracy is the total number of correct predictions divided by the number of predictions made on all datasets. In other words, this represents what percent of columns are appropriately tagged by machines for the labeled datasets. For instance, if we have 100 columns and K9 categories 90 of them correctly, the accuracy would be 90%. We cannot solely rely on accuracy, as columns with PII are a small percentage of all columns. A classifier that fails to classify anything would have an unusually high accuracy as the true negatives would outweigh the false negatives.
2. Precision: Precision is how "correct" our positive predictions are. The fewer false positives, the better our precision. However, it does not tell the full story. For example, if

we make a single prediction with a 100% success rate, but fail to categorize the other 9 sensitive columns, we have 100% precision, but low accuracy and low recall.
3. Recall (sensitivity): Recall is how likely we are to identify sensitive information. The fewer false negatives, the better our recall. However, it does not tell the full story. For example, if we predict all columns are sensitive information, we will get 100% recall, but low accuracy and low precision.
4. F2 Score: The F2 score is a way to skew towards optimizing recall over precision in a single measurable metric.

Precision vs. Recall

The impact of precision and recall would affect two different audiences. For example, low-precision metrics would be concerning for engineers and data scientists. Higher false positives would unnecessarily restrict access to non-sensitive data or delete them prematurely. If there are a lot of false negatives, the system might not be able to restrict sensitive data and enforce appropriate retention that would potentially violate the compliance contracts.

In addition to the above metrics, we strive to measure some additional metrics to track the engineering progress. These metrics measure the following perspectives:

1. Automation quality: We want to measure how many auto-tags are overturned by humans such as data owners/admins.
2. Scale: As we have to tag hundreds of thousands of datasets, we need to measure how many datasets we can onboard per day.
3. Re-classification: When to re-classify any dataset based on schema changes or when new tags are introduced/updated.
4. Efficiency: This initiative is based on data crawling, which is computationally expensive. We will track how much Uber pays to automate each data element (i.e., columns or tables).
5. Operation Excellence: Once the development process and initial onboarding are over, we will track how much operations overhead is needed, such as support, bug fixing, and on-call.
6. Storage System Coverage: As mentioned above, we have different storage technologies and backbones; tracking how many storage instances are onboarded to the categorization effort will be essential.

Production Experiences

Centralized Data Collection System

Uber employs a diverse range of storage systems across various infrastructures, each with its own unique characteristics. Some systems adhere to specific schemas, while others do not. Even among schema-based systems, there can be substantial variations in schema structures.

To streamline the categorization of data stored in these diverse systems, we have implemented a robust data collection system. This system samples data from different storage systems and consolidates them into a centralized data lake. Under this unified approach, the sampled data undergoes processing through a standardized workflow.

Key Benefits:

- Consistent Processing: By centralizing data into a common data lake, we facilitate the use of a standardized workflow for processing.
- Simplified Management: The approach simplifies the management of categorization jobs, offering a centralized point of control.

This strategy not only addresses the challenges posed by the varied nature of storage systems, but also enhances efficiency in data processing and management.

Advancements in Accuracy

Since its initial production launch, DataK9 has undergone continuous enhancements in accuracy over the past few years. We employ two key metrics to showcase the overall accuracy of DataK9 as illustrated in Figure 9:

- Accuracy for Golden Datasets: In this approach, we compare tagging results with golden datasets meticulously reviewed by our internal privacy experts. This metric reflects the precision and reliability of DataK9 against a standard set by privacy experts.
- Accuracy for Owner Reviewed Datasets: Additionally, we evaluate accuracy by comparing results with the categorization performed by the data owners. This metric provides insights into the alignment of DataK9 with the intended categorization as defined by those responsible for the data.

These metrics serve as robust indicators, illustrating the continual commitment to improving accuracy and checking DataK9's efficacy in meeting both internal privacy standards and the expectations of data owners.

Success Metrics & Funnel Analysis

In our pursuit of success, we have implemented a comprehensive set of metrics to gauge and optimize our automation process. A detailed funnel (see below) has been meticulously designed to facilitate the investigation and identification of gaps at each step. This invaluable tool provides a systematic approach to trace and monitor the overall categorization status, empowering us to make informed decisions and improvements.

Key Benefits:

- Granular Analysis: The funnel allows us to dissect the automation process into individual steps, enabling a granular analysis of performance at each stage.
- Gap Identification: By employing the funnel, we can effectively identify and narrow down gaps in the automation process, streamlining our efforts to enhance efficiency.
- Traceability: The funnel serves as a reliable tracking mechanism, offering real-time insights into the categorization status and allowing us to trace progress over time.

This meticulous approach to success metrics and funnel analysis reinforces our commitment to continuous improvement and enables us to proactively address challenges within our automation process.

Conclusion

The DataK9 project at Uber represents a pioneering effort to address the challenges of categorizing data at scale and at the field level through the implementation of AI and ML technologies. Recognizing the fundamental role of data categorization for privacy and security initiatives, Uber has undertaken this initiative to automate and streamline the process.

.