PikePlace: Generating Intelligence for Marketplace Datasets

Shi Qiao
SmartApps Inc.
shi@smart-apps.ai

Alekh Jindal
SmartApps Inc.
alekh@smart-apps.ai

ABSTRACT

There is a renewed interest in data marketplaces with cloud data warehouses that make sharing and accessing data on-demand and extremely easy. However, analyzing marketplace datasets is challenge since current tools for creating the data models are manual and slow. In this paper, we propose to demonstrate a learning-based approach to discover, deploy, and optimize data models. We present the resulting system, PikePlace, show an evaluation over Snowflake marketplace and TPC-H datasets, and describe several demonstration scenarios that the audience can play with.

PVLDB Reference Format:
Shi Qiao and Alekh Jindal. PikePlace: Generating Intelligence for Marketplace Datasets. PVLDB, 16(12): 4106 - 4109, 2023.
doi:10.14778/3611540.3611632

PVLDB Artifact Availability:
The source code, data, and/or other artifacts have been made available at https://pikeplace.smart-apps.ai.

1 INTRODUCTION

Data marketplaces are becoming increasingly popular for scenarios such as querying, reporting, data services, APIs, ML models, and views [12]. While traditional open-source marketplaces, such as Data.gov [9], OECD [14], World Bank [3], and Nikkei [13], have been around for a while, there is a renewed interest in data marketplaces with cloud data warehouses that make sharing and using data extremely easy. Consequently, several cloud data warehouse providers have launched their own data marketplaces, including Snowflake [17], AWS [2], Databricks [5], and Google [4]. Many other data-driven organizations also have similar data sharing internally, e.g., the Cosmos platform in Microsoft [15], where separate teams are responsible for producing and consuming enterprise data. Industry trends show that data marketplace platforms are growing at a CAGR of 25%, which is faster than both big data and business intelligence, and it is expected to be a $5$B$+ market by 2030 [16].

Data marketplaces users still need to analyze the datasets before deciding to buy or even spend time in using them. Unfortunately, such an analysis is ad-hoc and tedious, making it a challenge to get business value out of marketplace datasets [18]. To illustrate, consider the Amazon Vendor Analytics dataset from the Snowflake marketplace [1] that contains 52 tables and 1,568 columns. Typically, the data analysts start by running *"select * from T limit 10"*, before figuring out how to clean, combine, aggregate, and create interesting visualizations. However, this is tedious for the large number of tables and columns present in this dataset. In fact, analysts can easily end up spend days or weeks in wrangling marketplace datasets before they can derive the relevant insights [7]. Naturally, users want more efficient ways to onboard new external data source [8].

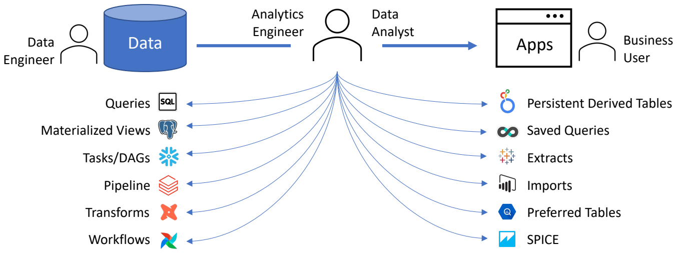


Figure 1: Current tools for manual data modeling.

The core challenge in turning data into intelligence is to create the right data models that can deliver insights quickly. Figure 1 shows popular tools used by data analysts (or the analytics engineers) for data modeling. Data analysts can either use data platform side tools such as writing queries directly to interactive data clouds, saving those queries as materialized views for predictive performance, scheduling them as tasks or DAGs in Snowflake or pipelines in Databricks, or running them as external workflows in DBT or Airflow. Alternatively, the data analyst can use application side tools such as persistent derived tables using LookML, saved queries in Apache Superset, Extracts in Tableau, Imports in Power BI, preferred tables in Google BI engine, and SPICE in Amazon QuickSight. Unfortunately, these tools are manual and complex, requiring a lot of time and effort to learn, build, and operationalize data models.

In this demonstration, we take a radically different approach to data modeling. Instead of having analysts figure out their data models manually, we introduce a model generator that automatically generates data models on marketplace datasets, as shown in Figure 2. With model generator, data analysts work on directly discovering the insights, while the system takes care of automatically sharing the models across all users. Once the generated models are deployed, the system also optimizes them for best performance, and composes them into scalable workflows. The resulting system, coined *PikePlace*, brings down the time to insights from days and weeks to minutes, while making it extremely easy to deploy and manage the data model lifecycle.

The rest of the paper is organized as follows. First, we present the PikePlace system and its various modules for data marketplace intelligence (Section 2). Then, we show a brief evaluation of PikePlace in terms of exploration effectiveness and performance (Section 3).

Proceedings of the VLDB Endowment, Vol. 16, No. 12 ISSN 2150-8097.
doi:10.14778/3611540.3611632

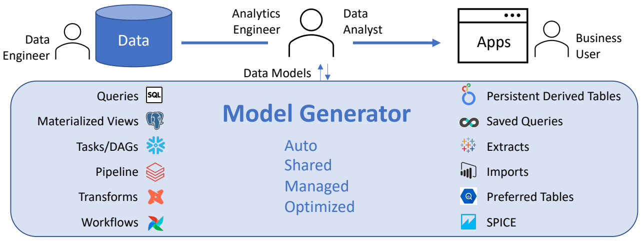


Figure 2: Generated data modeling approach.

And finally, we describe several demonstration scenarios and how the audience will play with them (Section 4).

2 PIKEPLACE

PikePlace introduces a brand-new way to think about marketplace intelligence, as illustrated in Figure 3. It starts by learning a data model generator for a given data source. Thereafter, user can ask for generated data models, explore interesting visualizations before tuning the data models further, and finally deploying them as generated workflows. With generated data models, PikePlace automates an important time-taking step, reducing the time and effort needed in gathering business intelligence. Finally, a workflow generator then deploys the data models by automatically creating optimized Airflow DAGs that run periodically on the backend.

Below we describe each of the components in PikePlace, namely the generated models, the exploratory visualizations, and the optimized workflows in more details.

2.1 Generated Data Models

Generating data models involves three steps, namely building the base models, inferring columns, and ranking final models. We describe each of these below.

Base models. PikePlace starts by connecting to users' data sources and learning the model generator directly over it. To do this, PikePlace first collects the schema information for each of the tables and views in the data source. For each table or view object, we gather various single and multi-column statistics using a sample on those objects. We could leverage different sampling techniques depending on the training budget. We also collect primary/foreign keys from each of the table and view objects and produce candidate denormalized base models for further processing.

Column inference. For each column in the base models, we apply quality checks to prune columns with too many missing values and de-duplicate highly correlated ones. We then infer columns as dimension, measure and filter columns based on their names, types, statistics, and data quality. We infer each column independently for each of the roles (dimensions, measures, filters) and the same column may be inferred for multiple roles. We train the parameters involved in column inference over different datasets and using the user response (selected or not) to each of the generated models.

Model ranking. Given base models along with their column inferences, we learn to rank a given model $M(b, d, m, f)$, where b is the base model, d is the dimension column, m is the measure column, and f is the filter column. The model ranking is in turn a function of individual rankings of base model and inferred columns.

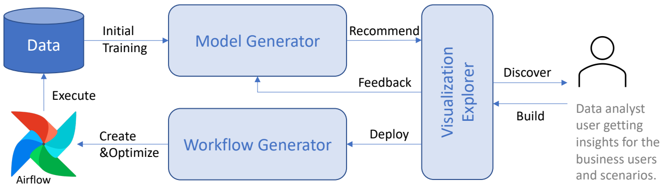


Figure 3: The PikePlace architecture.

We further incorporate typically seen model patterns across users and datasets as additional weights in the model ranking. We learn the initial ranking function using linear regression and manual labeling, but then use the user response afterwards to tune the parameters. Using better ML models for training will be part of future work.

The model generator enumerates valid data models, while prioritizing the typical patterns, and produces a ranked list of top-k models. Each model is assigned a model name and users can create new data models on top of existing ones.

2.2 Exploring Visualizations

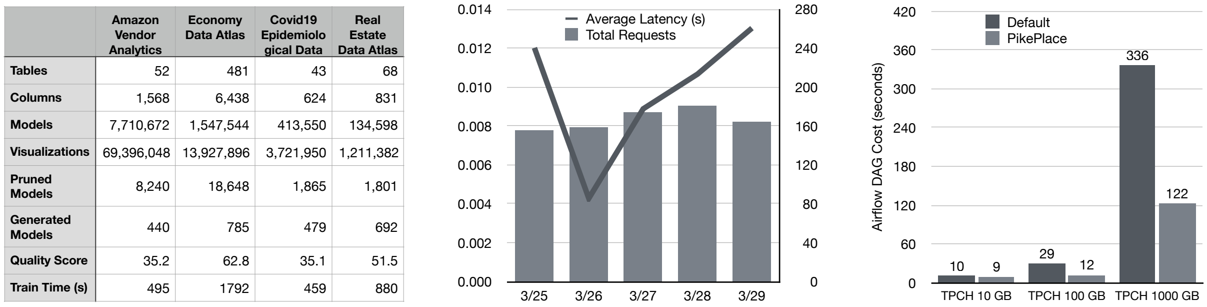
Once the model generator has trained on a data source, users can start exploring visualizations. There are three steps in this process.

Start exploration. A typical data analyst starts by selecting a subset of rows from each of the table/views in their data sources. They also plot the results to understand the shape and characteristics of the data. In contrast to this manual process, PikePlace allows users to directly start with interesting visualizations, ordered by their ranking, and then inspect the SQL statement, description, or usage of the data model behind them. Users can further refine the visualizations by editing any model's SQL statement. Thus, users get a better sense of what is available and useful in their data.

Similar models. The second part of the exploration allows users to see similar visualizations, i.e., either the same model presented differently or models that are very similar. This is still the discovery phase where the users are trying to figure out what other similar data models could be useful. Users can ask the model generator for as many new models as they want. The system will cache frequently explored models across all users for better performance and record the user activity for future training.

Search models. Finally, apart from the model recommendations, PikePlace also allows users to search the models. While many recent AI-driven approaches allow for natural language search over databases, the problem is to ensure correctness and usefulness of what is being generated. This happens because all those approaches focus on translating natural language to declarative SQL. Unfortunately, users need to provide carefully crafted prompts to get useful answers out of it. Instead, we learn from data to mine what are the useful models present in the first place, and then provide a guided search over those models. For each model generated by our model generator, we generate its natural language description from language models (using OpenAI's gpt-3.5-turbo), and then allow people to search over them using inverted indexes.

Overall, PikePlace simplifies the data modeling process by allowing users to explore and build on top of generated data models.



	Amazon Vendor Analytics	Economy Data Atlas	Covid19 Epidemiolo gical Data	Real Estate Data Atlas
Tables	52	481	43	68
Columns	1,568	6,438	624	831
Models	7,710,672	1,547,544	413,550	134,598
Visualizations	69,396,048	13,927,896	3,721,950	1,211,382
Pruned Models	8,240	18,648	1,865	1,801
Generated Models	440	785	479	692
Quality Score	35.2	62.8	35.1	51.5
Train Time (s)	495	1792	459	880

(a) Discovering data models.

(b) Interactive performance.

(c) Scalable operational costs.

Figure 4: PikePlace evaluation in terms of discoverability, performance, and cost.

2.3 Generated Workflows

Once the users are done with data modeling, they also want to deploy and keep their data models updated. PikePlace generates all required workflows for users' data models as described below.

Model folding. Users can save any of their models (recommended or modified) during the exploration phase. All saved models are deduplicated and collected into a single workflow. Since users can compose models on top of each other, PikePlace folds all models when adding them to the workflow. This means that model queries are automatically expanded using a series of WITH clauses. It is like query folding in Power Query [6]. Consequently, the backend query processing engine sees large query blocks that could be better optimized and pipelined by their optimizers and query processors, instead of seeing a series of small expensive queries.

Generate workflows. Instead of running live queries, PikePlace pushes down all data models into scalable offline workflows that could be run reliably. For example, a generated Airflow DAG with all the folded models. The DAG persists each model as a table or view that will be then used by visualizations selected by the user. All visualizations load instantly since they directly access pre-aggregated results. Users can specify the refresh rates for their charts and PikePlace takes care of running the workflow DAGs frequently enough to meet those requirements. We also check changes in the inputs to determine whether DAG executions could be delayed if the none of the inputs have any changes.

Optimize workflows. Instead of running each data model separately, PikePlace analyzes whether there are shared computations that could be reused across models to reduce the backend cost. Many production systems have computation overlaps in their production workloads and identifying or optimizing them manually is incredibly hard [10, 11]. PikePlace helps identify and share such overlaps automatically by inspecting the query trees and sharing common subtrees via materialized views. As a result, users can significantly reduce their computation costs, which is increasingly a concern in many enterprise settings.

3 EVALUATION

We now present a brief evaluation of PikePlace along three metrics that are relevant for business intelligence users:

Discovery. Figure 4a shows the discoverability of PikePlace on four different Snowflake marketplace datasets. These datasets have tens of tables, thousands of columns, and millions of model and visualization candidates. Yet, PikePlace can narrow down and generate few hundred high quality data models with a one-time training of a few hundreds of seconds.

Performance. Figure 4b shows the performance of interactive exploration of the same four Snowflake marketplace datasets with PikePlace, over a period of five days in March 2023. Each of these days have roughly 160 total requests from test users (all interactions across all datasets), and the average latency of these requests' ranges from 4ms to 13ms, with an average of 10ms.

Cost. Finally, Figure 4c shows the cost of running interactive analytics over TPC-H dataset, when scaling the dataset size from 10GB to 1TB. Users selected a variety of filter and aggregate models, and PikePlace pushed them down as an optimized Airflow job, which is significantly cheaper (workflow execution time 2.8x faster on 1TB) compared to running the default queries.

In summary, PikePlace helps analysts discover the right data models from massive search spaces, explore interesting visualizations at interactive speeds, and finally deploy them as optimized workflows with much cheaper cost.

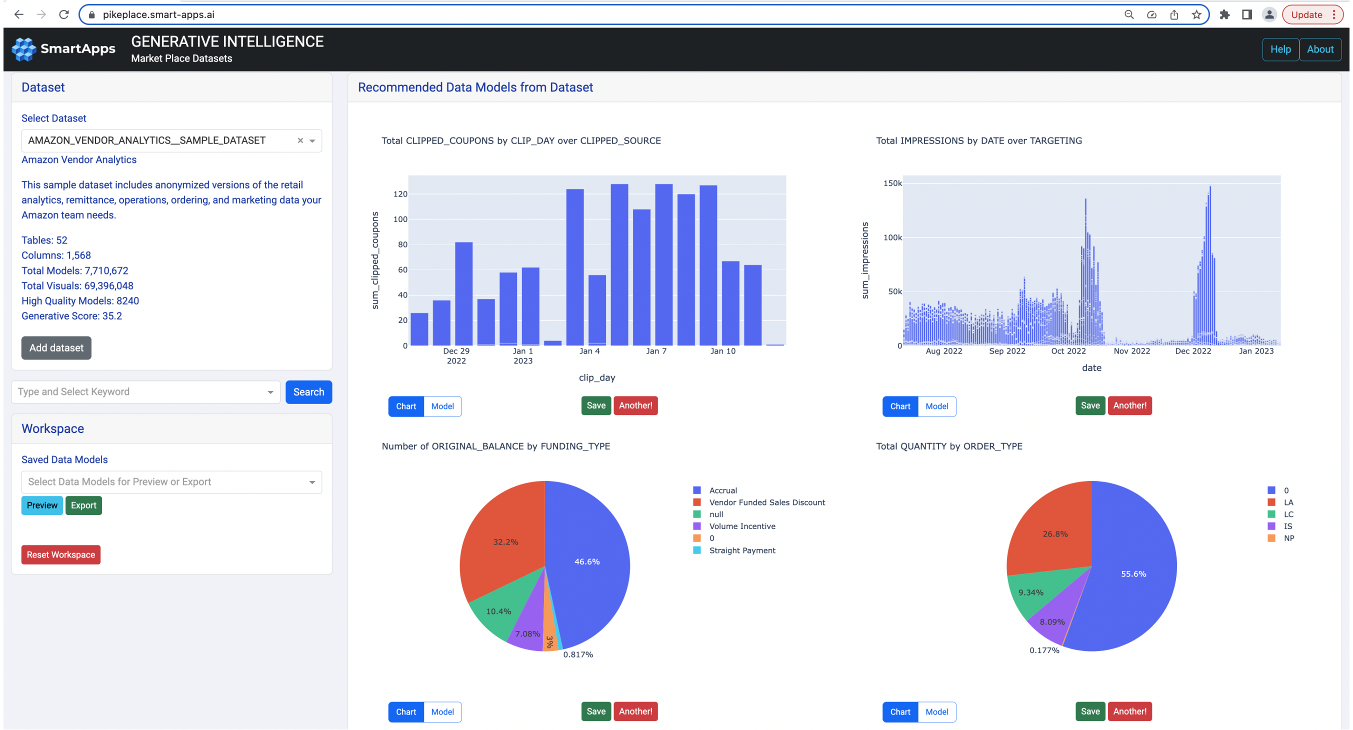
4 DEMONSTRATION

We now describe the demonstration setup and scenarios.

Setup. We will invite users to play with PikePlace using a AWS-hosted web portal, as shown in Figure 5. The portal will allow people to explore and gain insights over Snowflake marketplace datasets. We will pre-train on 20 most popular datasets, from different domains, to play with and the audience can also add new datasets that will train within a few minutes. The audience can watch and play on computer screens, and we will also provide a public link that anyone can open and play with on their mobile phones. Each user will have their own browsing session, which they can start and refresh anytime. Apart from free play, we will guide the audience through four concrete scenarios described below.

4.1 Dataset Exploration

First, users can simply point to a dataset and start exploring interesting explorations straightaway, without running any *'select * from T'* queries. For each dataset, users can see its overall summary, check out the most interesting model visualizations in there, and asl for other similar model visualizations. For each model, users



Select Dataset

AMAZON_VENDOR_ANALYTICS__SAMPLE_DATASET

Amazon Vendor Analytics

This sample dataset includes anonymized versions of the retail analytics, remittance, operations, ordering, and marketing data your Amazon team needs.

Tables: 52
Columns: 1,568
Total Models: 7,710,672
Total Visuals: 69,396,048
High Quality Models: 8240
Generative Score: 35.2

Type and Select Keyword

Workspace

Saved Data Models

Select Data Models for Preview or Export

Figure 5: Screenshot of the PikePlace web portal for marketplace intelligence.

can inspect its SQL statement, natural language explanation, sample usage, and its interestingness score. Users can also perform interactive actions like zoom, pan, select, or download the charts.

4.2 Guided Model Search

Once users are comfortable with a dataset, they can search for relevant concepts in that dataset. PikePlace provides a guided search, where all relevant keywords are extract, ranked, and indexed. Users can start typing and see which concepts exist, ranked by their score, and search relevant models for them. They can combine multiple concepts and ask for other similar model visualizations.

4.3 Model/Chart Workspace

As users discover useful models and charts, they can now start refining the model's SQL statement to their business needs, e.g., changing the measure or the chart type. They can save any useful model they find into the workspace while they continue exploring for more models and visualizations. They can also save models across different datasets, and anytime view one or more saved model visualizations together at once place.

4.4 Generating Optimized Workflows

Finally, once the users have explored and collected the required data models, users can now start operationalizing them for continuous updates. Users can export one or more saved models into their desired workflow formats, including Airflow, DBT, and LookML. Users can also check the SQL statements and the explanations of all selected models before exporting them for deployment.

REFERENCES

[1] Reason Automation. 2023. Amazon Vendor Analytics. https://app.snowflake.com/marketplace/listing/GZTYZ3HT1R1/reason-automation-amazon-vendor-analytics-sample-dataset
[2] AWS. 2023. Data Exchange. https://aws.amazon.com/data-exchange
[3] World Bank. 2023. Open Data. https://data.worldbank.org
[4] Google Cloud. 2023. Datasets. https://cloud.google.com/datasets
[5] Databricks. 2023. Delta Sharing. https://www.databricks.com/product/delta-sharing
[6] Miguel Escobar, Doug Klopfenstein, Jason Howell, and Peter Myers. 2022. Power Query query folding. https://learn.microsoft.com/en-us/power-query/power-query-folding
[7] John Farrall. 2023. Alternative Data Weekly #123. https://farrall.substack.com/p/alternative-data-weekly-123
[8] Forrester. 2023. External Data Sets. https://www.cruxdata.com/forrester-research
[9] U.S. Government. 2023. Open Data. https://data.gov
[10] Alekh Jindal et al. 2018. Computation Reuse in Analytics Job Service at Microsoft. In *SIGMOD*.
[11] Alekh Jindal et al. 2018. Selecting Subexpressions to Materialize at Datacenter Scale. *PVLDB* 11, 7 (2018), 800–812.
[12] Arvind Murali. 2021. The Ins And Outs Of A Data Marketplace. https://www.forbes.com/sites/forbestechcouncil/2021/05/12/the-ins-and-outs-of-a-data-marketplace
[13] Nikkei. 2023. Dataset. https://nkbb.nikkei.co.jp/en/dataset
[14] OECD. 2023. OECD Data. https://data.oecd.org
[15] Conor Power et al. 2021. The Cosmos Big Data Platform at Microsoft: Over a Decade of Progress and a Decade to Look Forward. *PVLDB* 14, 12 (2021), 3148–3161.
[16] Grand View Research. 2022. Data Marketplace Platform Market Growth & Trends. https://www.grandviewresearch.com/press-release/global-data-marketplace-market
[17] Snowflake. 2023. Snowflake Marketplace. https://www.snowflake.com/en/data-cloud/marketplace
[18] Matei Zaharia, Zaheera Valani, Jay Bhankharia, Sachin Thakur, Itai Weiss, and Steve Mahoney. 2022. Introducing Databricks Marketplace. https://www.databricks.com/blog/2022/06/28/introducing-databricks-marketplace-an-open-marketplace-for-all-data-and-ai-assets.html