International Journal on Science and Technology (IJSAT)
E-ISSN: 2229-7677 ● Website: www.ijsat.org ● Email: editor@ijsat.org

Optimizing Snowflake Enterprise Data Platform Cost Through Predictive Analytics and Query Performance Optimization

Shreesha Hegde Kukkuhalli

hegde.shreesha@gmail.com

Abstract

The rapid adoption of cloud-based data platforms, such as Snowflake, has led to significant benefits in terms of scalability, flexibility, and performance for modern enterprises. However, managing costs in such environments remains a challenge, especially as data volumes and query complexities increase. This paper explores a comprehensive strategy to optimize Snowflake costs through the implementation of predictive analytics and performance optimization techniques. By leveraging machine learning models to forecast resource utilization and employing query optimization techniques, organizations can reduce operating expenses without compromising performance. The results from experiments demonstrate a significant reduction in costs and improved system efficiency.

Keywords: Snowflake, Cloud Cost Optimization, Predictive Analytics, Performance Tuning, Enterprise Data Management, Machine Learning, Query Optimization, Cloud Computing.

I. Introduction

Cloud-based data platforms, such as Snowflake, have become integral to enterprises seeking scalability and high-performance data processing solutions. These platforms offer virtually unlimited scalability, offload infrastructure maintenance and database administration, enabling businesses to handle vast amounts of data. However, as data grows, so does the cost of operations. Unoptimized queries, inefficient storage utilization, lack of governance and underutilized resources lead to increased financial expenditures, often exceeding allocated budgets.

Problem Statement: As enterprises continue to scale, the cost of operating on platforms like Snowflake increases significantly. This paper investigates how to optimize costs on Snowflake using predictive analytics and performance optimization techniques. The goal is to reduce the overall cost of operating Snowflake environments while maintaining or improving performance.

Objectives: The key objectives of this paper are:

1. To identify cost drivers in the Snowflake environment.
2. To explore predictive models for forecasting usage patterns and costs.
3. To implement query and performance optimization techniques for improved cost-efficiency.

II. Main Body

Snowflake Overview: Snowflake is a cloud-native data warehouse solution that provides a single platform for all data warehousing needs. It allows for the storage, management, and querying of structured and semi-structured data across multiple cloud service providers. Snowflake's ability to scale elastically and its pay-per-use pricing model make it a popular choice among enterprises. However, without careful cost management, organizations may face unexpected charges as workloads and usage grow.

Cloud Cost Management: Several approaches to cloud cost management have been proposed. Traditional methods rely on manual resource management, such as resizing instances and monitoring usage. Recent advances focus on leveraging automation and predictive analytics to preemptively scale resources and optimize costs dynamically.

Performance Optimization to Optimize Cost Savings: Performance optimization in Snowflake often involves tuning queries, optimizing the use of virtual warehouses, and efficiently managing storage. Techniques like caching, partitioning data, and leveraging materialized views have been widely used to enhance system performance and reduce costs. However, integrating predictive analytics for dynamic cost optimization has not been extensively studied.

Methodology

Identifying Key Cost Drivers in Snowflake: To optimize Snowflake's cost, it is critical to first understand the factors contributing to its costs. Snowflake follows pure consumption based pricing. Key factors include:

- **Virtual Warehouse Costs**: Snowflake charges based on the size and duration of virtual warehouse usage for compute. Snowflake provides multiple warehouse types such as x-small, small, medium, large, x-large, compute credit which in turn cost doubles with each higher warehouse as shown below. Virtual warehouse costs typically form majority of overall snowflake cost.

Warehouse Type	Credit	Cost/hour
x-small	1	$4
Small	2	$8
Medium	4	$16

- **Storage Costs**: The amount of data stored in Snowflake influences overall costs. Storage cost typically accounts for a small part of overall snowflake cost.
- **Data Transfer**: Data egress and transfer between regions or cloud providers also incur substantial costs.
- **Query Complexity and Execution Time**: Complex queries that require more compute resources and execute for a longer duration, drive up warehouse usage and costs.

- **Cloud services**:cloud services layer of the Snowflake architecture consumes credits as it performs behind-the-scenes tasks such as authentication, metadata management, and access control.
- **Serverless compute**:Cost associated with Snowflake features such as Search Optimization and Snowpipe that use Snowflake-managed compute resources rather than virtual warehouses.

Predictive Analytics Model for Cost Forecasting: The core of the cost optimization strategy involves building a predictive analytics model to forecast future costs based on historical usage patterns. This model will help identify cost spikes and trends, enabling proactive cost management.

Data Collection

The data for the model includes metrics such as:

- Virtual warehouse usage (e.g., compute time, query load).
- Storage volume trends over time.
- Query execution time and complexity metrics.
- Data transfer and replication logs.

Sample query to extract warehouse cost for year2024

```
select warehouse_name, round(sum(credits_used)*4,2) as warehouse_cost from snowflake.account_usage.warehouse_metering_history where start_time>= '2024-01-01' and start_time<= '2024-12-31' group by warehouse_name order by sum(credits_used) desc;
```

Model Selection and Training

Several machine learning models are considered for predicting cost and resource usage, including:

- **Linear Regression**: Suitable for identifying linear relationships between resource usage and costs.
- **XGBoost**: A more robust model that can handle complex, non-linear relationships in the data.

Sample code to predict usage based on historical data using XGBoost

```
def workload_predictor(historical_data):
    features = extract_temporal_features(historical_data)
    model = XGBoostRegressor(
max_depth=6,
learning_rate=0.1,
n_estimators=100
    )
    return model.fit(features, historical_data['resource_usage'])
```

- **Time Series Forecasting**: Autoregressive Integrated Moving Average (ARIMA) and Prophet models are explored to forecast resource usage based on historical trends.

The model is trained using historical usage data from Snowflake, and the performance is evaluated based on the accuracy of cost predictions and early identification of potential cost overruns.

Query and Performance Optimization

To complement the predictive analytics model, performance optimization techniques are applied to reduce overall resource consumption:

- **Query Refactoring**: Queries are analyzed for inefficiencies, and optimizations such as reducing the number of joins, limiting data scans, and utilizing partition pruning are implemented.
- **Caching and Clustering**: By leveraging Snowflake's result caching and data clustering features, query execution times are reduced, lowering compute costs.
- **Search optimization service**:Enable this feature to improve the performance of certain types of look up and analytical queries.
- **Virtual Warehouse Sizing and Scaling**: Dynamically adjusting the size of virtual warehouses based on forecasted usage helps avoid over-provisioning, thus reducing compute costs.
- **Materialized Views and Indexed Data**: Using materialized views for frequently queried data sets significantly reduces the workload on virtual warehouses.

Case Study

Background:A large manufacturing firm had snowflake deployed across its four operating regions: Americas, Emea, Apac and China, and snowflake was the primary component of company's enterprise data platform. Firm was seeing increase in snowflake compute cost month over month along with difficulty in predicting the cost resulting inaccurate budget forecast. Total snowflake cost exceeded $1M per year and more than 90% of the snowflake cost was associated with virtual warehouse compute cost, rest of the cost was associated with other services such as storage, serverless features etc.

Firm wanted a reliable way to predict snowflake consumption cost along with finding opportunities to optimize cost.

Snowflake consumption chart

Implementation: Following key steps are followed to build the predictive model and identify opportunities to optimize queries to save cost.

- 12 months of historical data was collected from snowflake account usage and query statistics tables.
- Data was analyzed for following performance optimization opportunities:
  - Large tables that are never queried
  - Tables where data written but not read
  - Rarely used materialized views
  - Short-lived permanent tables
  - Rarely used search optimization paths
  - Rarely used tables with automatic clustering
  - Inefficient usage of multi-cluster warehouse auto scale
  - Very large tables with no clustering key defined
  - Opportunity to use query caching
  - Share virtual warehouses across different teams/workloads
- Time series forecasting model was built based on historical data collected over last 12 months. Parameters are adjusted to ensure model accuracy is > 95%.

Results and Benefits

- With the help of performance optimization techniques, on an average running time of long running queries was decreased by ~30% resulting 30% reduction in cost incurred to run these queries.
- Time series model reliably predicted the future usage and cost, allowing the teams to optimize virtual warehouse scale in and scale out strategy saving 10% in compute cost over 6 months validation period.
- Resource monitors are set to prevent accidental spike in usage due to incorrect virtual warehouse usage or unoptimized queries.

Discussion

The results of this study demonstrate the effectiveness of using predictive analytics to forecast Snowflake resource usage and costs. By incorporating machine learning models and combining them with performance optimization strategies, significant cost reductions can be achieved. However, there are limitations, particularly in environments with highly unpredictable workloads, where predictive models may need continuous adjustment to maintain accuracy.

Additionally, performance optimization techniques need to be tailored to specific query patterns and data structures. A one-size-fits-all approach may not work across all workloads, and organizations may need to invest time in understanding their data consumption patterns.

III. Conclusion

Optimizing costs in cloud-based data platforms such as Snowflake is essential as enterprises continue to scale. By using predictive analytics to forecast resource usage and implementing performance

optimization techniques, organizations can achieve significant cost savings while maintaining or improving system performance. Future work will focus on refining machine learning models for more dynamic environments and exploring additional optimization techniques, such as automatic workload balancing and real-time query optimizations based on dynamic query execution plan and table statistics.

References

1. G. Smith, J. Brown, *Cloud Cost Optimization: Strategies for the Enterprise*, IEEE Transactions on Cloud Computing, vol. 10, no. 3, 2022.
2. M. Taylor, S. Allen, *Predictive Analytics in Cloud Resource Management*, Journal of Cloud Computing, vol. 8, no. 1, 2021.
3. Snowflake Inc., *Snowflake Architecture and Performance Optimization Guide*, 2023.
4. S. Roberts, *Machine Learning for Predictive Cost Forecasting in Cloud Environments*, IEEE Big Data Conference, 2021.
5. K. Jain, M. Kapadia, *Cost Management in Cloud Data Warehouses: An Optimization Perspective*, IEEE Cloud Computing, vol. 7, no. 1, pp. 40-51, 2020.
6. Snowflake Inc., *Snowflake Best Practices for Query Optimization and Cost Reduction*, White Paper, 2022.
7. Snowflake documentation reference: https://docs.snowflake.com/en/user-guide