Predictive Analysis of Client Applications on Multi-cloud to Optimize Cost

MSc Research Project
Cloud Computing

Muhammad Osama Khan
Student ID: x21242887

School of Computing
National College of Ireland

Supervisor: Diego Lugones

National College of Ireland
Project Submission Sheet
School of Computing

Student Name:	Muhammad Osama Khan
Student ID:	x21242887
Programme:	Cloud Computing
Year:	2023
Module:	MSc Research Project
Supervisor:	Diego Lugones
Submission Due Date:	15/08/2023
Project Title:	Predictive Analysis of Client Applications on Multi-cloud to Optimize Cost
Word Count:	6090
Page Count:	22

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	Muhammad Osama Khan
Date:	14th August 2023

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	☐
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	☐
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	☐

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Predictive Analysis of Client Applications on Multi-cloud to Optimize Cost

Muhammad Osama Khan
x21242887

Abstract

A key challenge for enterprises in the era led by cloud services is maintaining client application efficiency in costs, especially on multi-cloud platforms. In this paper, a predictive analytical approach for cost-effective application operation across AWS, Azure, and GCP platforms is introduced. Four-year billing patterns for client applications were predicted through data collection and simulation on actual billing patterns from e-commerce event data. This innovative approach not only predicts these prices but also takes it a step further by incorporating a cost optimization technique that deliberately makes use of the most cost-effective offerings among the three cloud providers. This approach will increase the efficiency of resource allocation and reduce operational costs in multi-cloud systems by up to 34%. Additionally, businesses may continuously improve their cloud plans, ensuring continued alignment with their financial goals, by evaluating and comparing the predicted data with actual operational bills on a monthly basis. In the rapidly changing computing environment, this agility, together with the insights provided by the study, opens the way for more economically and environmentally sound cloud implementation.

1 Introduction

The landscape of business computing has been drastically changed by the growth of cloud computing. Multi-cloud environments are the height of this advancement, offering scalable, on-demand, and cost-effective resources through cloud-based solutions. Utilizing multiple cloud providers enables companies to take advantage of the unique benefits and features of each service. Although multi-cloud solutions provide flexibility, robustness, and adaptability, successfully controlling costs and maintaining optimal utilization of resources become key challenges.

The ongoing issue of cost control is central to the discussion of multi-cloud setups. While implementing multi-cloud methods has many advantages, it also presents challenges in negotiating the many pricing plans, invoicing schemes, and reporting tools that are particular to each provider. Because of this dynamic, companies need to monitor resource use carefully and use advanced analytical techniques.

Data transport expenses, which are frequently disregarded, can significantly increase the overall cost of multi-cloud systems. Alshammari et al. (2017) go into detail on the subtleties of optimizing resource usage to successfully reduce costs. Businesses must do complex analyses to guarantee cost-effectiveness since each supplier has a different pricing architecture.

This paper aims to explore these differences in great detail. The importance of cost optimization in multi-cloud setups grows as firms strive for competitive advantages. Effective resource management reduces operating costs while also improving application performance, a crucial aspect in today's fast-paced corporate environment.

This study acknowledges major advancements achieved in understanding cloud resource prices, pricing structures, and cloud provider selection by drawing on a wide range of studies, including Osypanka and Nawrocki (2022), Ramesh et al. (2021), and others. However, there is a clear gap when it comes to predicting client application behavior.

According to the research, a carefully designed predictive analytic model may significantly improve cost optimization while maintaining application performance criteria. We provide a ground-breaking process that includes thorough data collection from several cloud platforms. An in-depth analysis will be performed on this data in order to identify patterns and trends. The ultimate outcome of this approach will be the development of a predictive model that offers cost-optimization strategies and forecasts the utilization of resources.

The expected results indicate that this approach can evaluate historical multi-cloud platform data to discover major cost-saving possibilities. In section 6 we provide an accurate analysis of our method in terms of resource consumption and cost predictions.

The efficient orchestration of resources is a key challenge, along with financial considerations. This perspective is best expressed by Rossi (2017), who emphasizes the regularity and rapidity of software deployments. For enterprises with limited IT resources, understanding and coordinating the range of tools, platforms, and services among cloud providers becomes a tough effort.

However, there are several possible benefits. Better service delivery, optimal application performance, and improved user experience are all results of efficient resource management.

Furthermore, multi-cloud environments face several other challenges in addition to those relating to cost and resource use. Because each cloud platform has its own set of tools, APIs, and data formats, integration can be challenging. Security, which is always of the utmost importance, differs among suppliers, calling for a consistent security approach. Additional challenges that enterprises must overcome include performance optimization, data management, and promising data synchronization between platforms.

The implications of this research are relevant to companies going through digitization efforts at optimal cost while leveraging the advantages of multi-cloud infrastructures. Implementing this predictive model lays the way for wise decision-making, precise cloud resource allocation, and improved user experience. Understanding and optimizing cost and resource usage in the context of multi-cloud systems could bring in a new era of productivity and expansion.

2 Related Work

As businesses increasingly adopt multi-cloud strategies to cut costs and improve their services, there has been a surge of interest in predictive analysis of client applications on multi-cloud platforms. This survey of the literature will offer a critical comparison of earlier studies on the topic, stressing the main challenges, goals, and contributions in the area. Predictive modeling, cost optimization potential, data collection, transformation, and analysis will all be covered in the literature study.

2.1 Cost optimization

An innovative method for cloud resource cost optimization that uses anomaly detection, machine learning, and particle swarm optimization is presented by Osypanka and Nawrocki (2022). Their approach is to forecast resource consumption and modify cloud components accordingly. It does not, however, specifically address the difficulties associated with resource demand forecasting and cost optimization in the context of multi-cloud environments like AWS, Azure, and GCP. By merging data gathering methods from different settings and utilizing the most recent developments in machine learning, this project aims to construct a predictive analytic model for cost optimization in client applications across several cloud providers.

Ramesh et al. (2021) research focuses on comprehending the economics of cloud computing, including pricing models and overall cost structures. Despite discussing a variety of pricing methods and cost structures, the article does not specifically address how to maximize resource efficiency and cost-effectiveness in a multi-cloud system. This study fills this knowledge gap by emphasizing the prediction of client application behavior in a multi-cloud environment to assist businesses in making the best use of their resources and maximizing their cost-effectiveness.

For multi-cloud systems, Tang (2022), provides a fault-tolerant cost-efficient workflow scheduling algorithm (FCWS) that takes into account system dependability, different payment mechanisms, and the heterogeneity of virtual resources. Through the development of a more precise task execution reliability evaluation approach and the consideration of a wider range of billing mechanisms, including complicated billing structures, this research seeks to decrease application execution costs while assuring reliability.

In conclusion, current research has made significant improvements in understanding and optimizing cloud resource costs, pricing models, and cost structures. Examples are Osypanka and Nawrocki (2022), Ramesh et al. (2021), Tang (2022), Kumar et al. (2019) and Horn et al. (2019). These studies do not, however, specifically address the difficulties of multi-cloud systems or the potential advantages of integrating predictive analytic techniques for cost optimization in client applications. This paper aims to close this gap by creating a predictive analytic model that takes into account a variety of billing systems and focuses on forecasting client application behavior across several cloud providers. This research seeks to provide a comprehensive and adaptive solution that enables businesses to optimize the utilization of resources and achieve improved cost efficiency in multi-cloud environments

2.2 Cloud Provider Selection

Heilig et al. (2020) addresses the Cloud Service Purchasing Problem (CSPP) in multi-cloud systems, concentrating on choosing the best virtual machine configurations for hosting application activities and lowering virtual machine purchase prices. Without taking into account any future changes in customer preferences or application needs, the suggested solutions concentrate on reducing costs and enhancing performance based on present requirements. In contrast, this research incorporates predictive analysis into the decision-making process for multi-cloud configurations to better forecast the demands of client applications in the future and change resource allocation appropriately.

Pandey et al. (2022) introduced OnTimeURB, a multi-cloud resource broker that helps customers choose the best cloud resources based on performance, agility, cost, and security (PACS) considerations. They offer the best cloud template solutions for diverse bioinformatics application processes across four main CSPs, comprising more than 300 alternative instance configurations, using integer linear programming and a Naive Bayes classifier. The evaluation's findings show that OnTimeURB performs much better in terms of cost-effectiveness and agility than a cutting-edge k-nearest neighbors (k-NN) method. Additionally, compared to methods without knowledge-engineered multi-CSP resource brokering, it improves process execution times with a success rate of 91 percent. Although Pandey et al. (2022) deals with the issue of selecting the best resources across various cloud service providers, neither of these issues is explicitly addressed in their work, which also does not make use of predicting the behavior of client applications. Hence, there is a need for more research in this area.

Georgios et al. (2021a) explored hybrid cloud options, focusing on effectiveness and cost savings in multi-cloud setups, especially for IaaS. They compare the pricing policies of 23 different providers, and then, using Data Envelopment Analysis (DEA) technique, they assess the efficiency of multi-provider service bundles. The study finds that multi-cloud solutions are more effective and may result in cost savings. Despite these results, the study has limits because it does not look at forecasting client application behavior for additional cost optimization.

IaaS pricing is clarified by Belusso et al. (2018) using a linear regression model that takes data center locations into consideration. Instead, this study uses historical event data to forecast and optimize billing across many cloud platforms with a focus on long-term efficiency and savings. For the most optimal VM allocation in cloud data centers, Zhang et al. (2020) recommends the Improved Differential Evolution (IDE) strategy, which improves both user task completion and provider costs. This study, in contrast, targets cost predictability and savings to estimate and optimize costs across many cloud platforms, including AWS, Azure, and GCP.

Predictive analysis for cost reduction across many clouds is the main focus of this paper. However, in order to increase efficiency, Belgacem (2022) explores dynamic resource allocation in cloud computing. Both deal with cloud optimization, but from different perspectives.

This research emphasizes the use of predictive analysis to cut expenses by examining client apps running on several cloud platforms (AWS, GCP, Azure). In contrast, Megahed et al. (2019) focuses on creating the best cloud solutions from the service provider's standpoint, assuring minimal costs while meeting client needs. Singh et al. (2019) offers an adaptive prediction model using support vector regression, ARIMA, and linear regression for shifting web application demands. A workload classifier chooses the right model depending on the unique features of the task. The findings indicate that web application service quality and error metrics have significantly improved in cloud settings.

Di Modica et al. (2019) describes the difficulties that businesses have while using multiple cloud environments, especially as a result of configuration and administration issues. The primary goal is to investigate the effects of different application deployment strategies on elements such as service quality, security, and provisioning costs in multi-cloud systems, with the help of experiments on actual cloud platforms. The issue of Web application replication and deployment in multi-cloud contexts is addressed in Shi et al. (2021). It presents the MCApp technique, which improves user request routing and application deployment to save costs. Using Data Envelopment Analysis (DEA), Georgios et al. (2021b) examines the effectiveness and cost of multi-cloud solutions among 23 IaaS providers. By examining both functional and non-functional features of IaaS services, the paper emphasizes the significance of cost-efficiency in cloud solutions.

Addya et al. (2021) offers CoMCLOUD, a broker model that balances cost and QoS while optimizing the deployment of multi-tier applications among multiple cloud providers. Alonso et al. (2023) evaluates the transition from using a single cloud service to using multiple clouds in an organized way, describing multi-cloud native apps and examining their effects on the development, deployment, and cost. The cost savings achieved by the flexible usage of cloud resources for Computational Intelligence (CI) applications have been highlighted by Horn et al. (2019). It demonstrates the difficulty of manually maintaining cloud resources across service providers such as AWS, GCP, and Azure and suggests the MELODIC platform as a way to reduce cross-cloud deployment costs. Sangeetha et al. (2022); Heilig et al. (2016); de Carvalho et al. (2018); Mishra et al. (2020) also discussed optimized resource allocation strategies.

The approach mentioned in this paper explicitly addresses the predictive analysis of client applications on multi-cloud platforms (AWS, GCP, Azure), whereas the studies covered previously mostly concentrated on generic cost-saving tactics and dynamic cloud resource allocation.

3 Methodology

The research methodology consists of 5 stages as shown in Figure 1 Data Gathering, Data Generation for Client Application, Prediction, Prediction with Cost Optimization, Evaluation, and Tracking Billing on a Monthly Basis. The subsequent sections provide a concise explanation of each stage.

3.1 Stage 1: Data Gathering

Companies rarely disclose data about cloud costs and budgets. So, in this research, we have created a methodology to synthesize a large dataset based on real available information in the public domain and real billing patterns from existing cloud providers. This predictive billing data was built on e-commerce event history data that was made accessible by Kaggle *eCommerce events history in electronics store* (2021). The initial dataset was just 5 months long, but by programmatically reproducing the data patterns, we could extend it to 2 years. Although simple, this technique gave us sufficient data over longer periods to conduct our investigation. The dataset has also been transformed in order to meet the criteria of the research in terms of formatting and presentation. The data was modified to show hourly metrics after being initially displayed at short intervals. To replicate a more demanding usage pattern, we also increased the number of events each hour and expanded the dataset. With this improvement, the data was much better and now more accurately represented a realistic client application scenario.

3.2 Stage 2: Data Generation for Client Application

The initial input of the data creation process was the generated data from the previous stage. We created a scenario where the application runs in a multi-cloud environment using Azure Compute, AWS, and GCP services, as shown in Figure 7. The algorithm was developed to generate data that was identical to the event patterns and pricing for associated cloud services which were provided. As a result, a thorough two-year dataset was produced, which was utilized as historical data for the research.

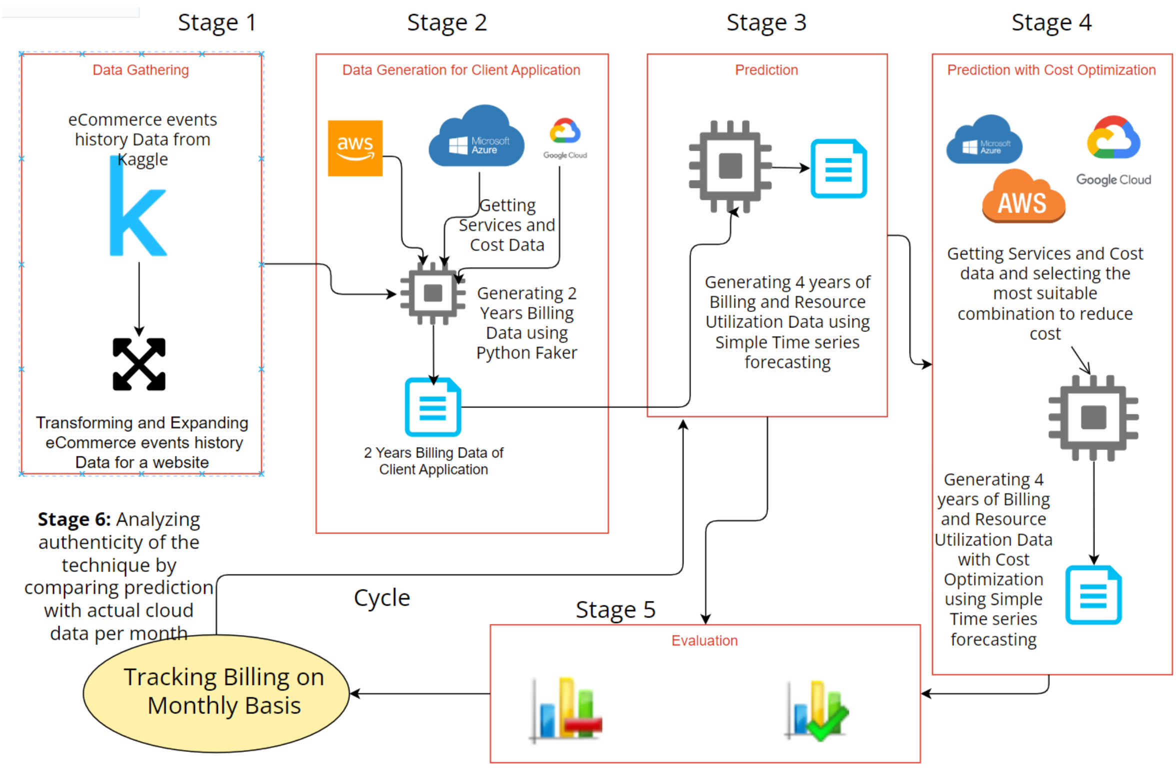

Figure 1: Predictive Analysis of Client Applications on Multi-cloud to Optimize Cost Stages

3.3 Stage 3: Prediction

We used 2 years of multi-cloud application billing and resource utilization data to predict costs for the next 4 years. A 1% annual increase in the cost of the cloud provider and a 20% annual growth in the client's application utilization were included as extra inputs to our simple time series forecasting model.

3.4 Stage 4: Prediction with Cost Optimization

We used an innovative approach for predicting during the cost-optimization stage of our study, taking on the findings and methods from Stage 3. The goal of achieving cost effectiveness without sacrificing performance became more and more clear as we looked further into forecasting the billing data for the future four years. In this stage, the most suitable service combination was carefully selected. based on the client application, Figure 3 shows cost details of competing cloud providers in relation to the application requirements. Cheaper substitutes from alternative providers were chosen at the detailed level. However, we optimized further and observed that certain compute instances frequently utilized less than 60 percent of their capacity after analyzing resource utilization trends. Recognizing the opportunity, we moved these instances to less expensive resulting in cheaper types while still meeting performance requirements.

The system then projected the optimal billing data over a four-year period after implementing these cost-saving measures. The results of this study showed how the cost-optimization strategies actually saved money, which is in line with the main goal of our research, achieving reliable multicloud operational efficiency.

3.5 Stage 5: Evaluation

In the evaluation stage, the primary objective was to compare the prediction data produced by Stages 3 and 4. To measure the effectiveness of the cost-optimization strategies and to provide an understanding of the cost effects, this comparison study was crucial. The data generated during Stage 3 was used as our starting point and provided an estimate of the client's application expenses while utilizing their current set of services. On the other hand, Stage 4 gave us a prediction that took into consideration the proposed strategic cost-saving measures. The estimated cost was a result of carefully selected cloud provider services that matched performance requirements while being more affordable.

Bar charts were created that represent the total monthly expenses for each step in order to visually demonstrate and compare these forecasts. The comparison was interesting because it demonstrated how real savings may be achieved by customers who use a variety of cloud providers. The difference between the two highlighted both the potential cost savings and the significance of ongoing assessment and optimization in multi-cloud setups.

3.6 Stage 6: Tracking Billing on Monthly Basis

Comparison of the predictions made by the predictive model with the newly-emerging actual monthly operational billing data is mandatory to confirm its correctness and usefulness. The model will be gradually enhanced in case there are differences between the forecasts and the actual expenses. This strategy not only confirms the accuracy of the forecasts but also allows it to adapt to any unexpected operational or financial changes.

4 Design Specification

In order to simulate the actual use cases of multi-cloud apps, we created a model based on a multi-cloud architecture, which includes Azure Compute services, GCP and AWS. We used an e-commerce event history dataset as the basis for our design and altered it to create a simulation of two years' worth of billing information. Our predictive billing system used this previous information to forecast the client's billing over the next four years. We used a cost optimization approach to improve our forecasts and provide benefits. The goal of this approach was to propose a solution that might significantly reduce costs by utilizing the optimal combination of cloud services, giving precedence to cost-effective solutions, and modifying cloud instance types according to their actual utilization of resources.

4.1 Workflow Diagram

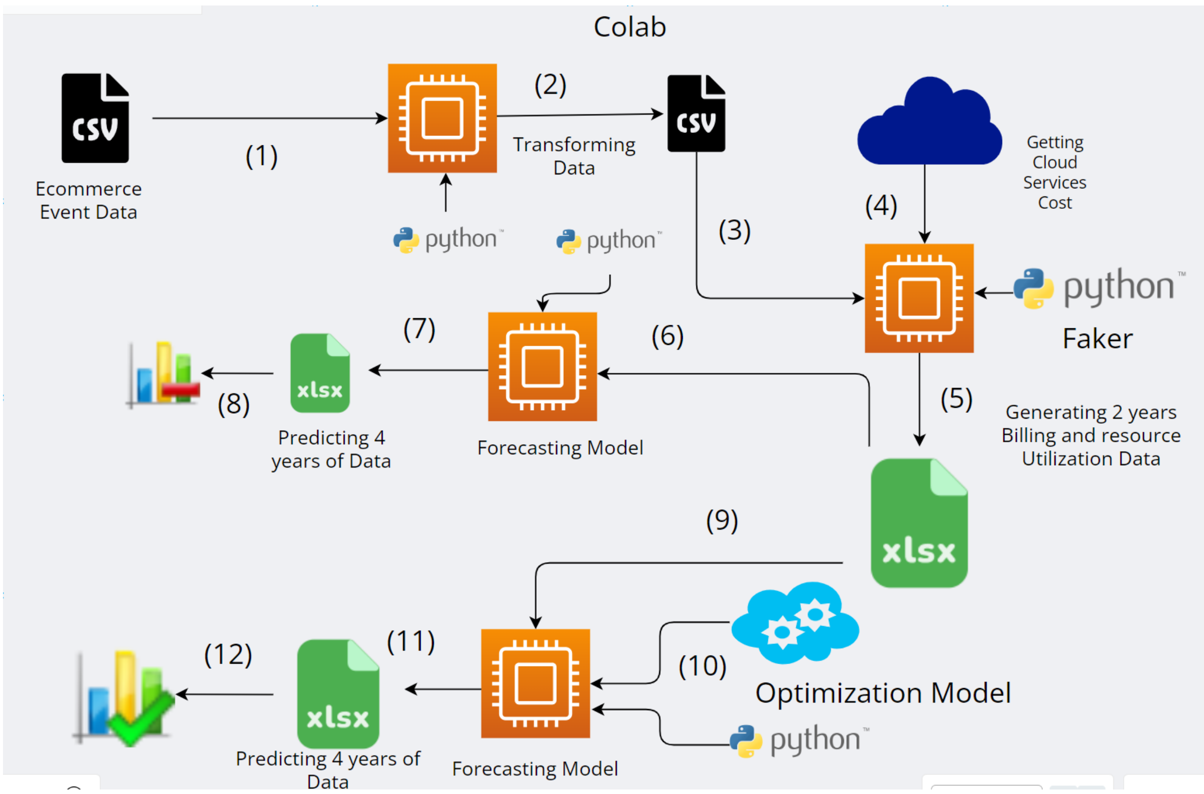

Figure 2: Workflow Design

The first step, as shown in Figure 2 is processing E-commerce Event Data using a Python script, converting it to the format we want, increasing its volume, and then creating a new file as shown in step 2. By Step 3, this new file which contains hourly event data lays the foundation for the generation of two years' worth of billing data, taking into consideration the particular expenses related to different cloud services, as specified in Step 4. In Step 5, which helps produce two years' worth of billing and utilization of resources data, the Python faker package is used. A Python-based forecasting algorithm then analyses this two-year billing data in Step 6 after that. This model makes a four-year billing and utilization of resources forecast for the client's application in Step 7. For simpler comprehension, Step 8 uses a bar chart to depict this complex data. The steps from Steps 6 to 8 are repeated in Steps 9 until 12. However, Step 10's incorporation of a cost optimization model illustrates an important difference. By offering the most cost-effective service combinations for the client's application, this approach optimizes the forecasting process and ensures more affordable forecasting solutions.

4.2 Cost Optimization Metrix

AWS Service	AWS Cost	Azure Service	Azure Cost	GCP Service	GCP Cost
EC2 t2.micro	$0.01260000	B1 Series	$0.01600000	f1-micro	$0.00900000
EC2 t2.small	$0.02500000	B1ms Series	$0.02800000	g1-small	$0.03000000
EC2 t2.medium	$0.05000000	B2s Series	$0.05600000	n1-standard-1	$0.05500000
RDS db.t2.micro	$0.01800000			db-f1-micro	$0.01050000
RDS db.t2.small	$0.03600000			db-g1-small	$0.03500000
RDS db.t2.medium	$0.07300000			HA db-g1-smal	$0.07000000
S3 GET	$0.00040000	Blob Storage GET	$0.00036000	Cloud Storage GET	$0.00032000
S3 PUT	$0.00050000	Blob Storage PUT	$0.00045000	Cloud Storage PUT	$0.00040000
S3 COPY	$0.00050000	Blob Storage COPY	$0.00045000	Cloud Storage COPY	$0.00040000
S3 POST	$0.00050000	Blob Storage POST	$0.00045000	Cloud Storage POST	$0.00040000
S3 LIST	$0.00500000	Blob Storage LIST	$0.00450000	Cloud Storage LIST	$0.00400000
DynamoDB Write request unit	$0.00065000	Cosmos DB Write RU	$0.00079950	Firestore Write	$0.00065000
DynamoDB Read request unit	$0.00013000	Cosmos DB Read RU	$0.00015990	Firestore Read	$0.00013000
Lambda Request	$0.00000020	Functions Request	$0.00000016	Cloud Functions Req	$0.00000014
SQS Request	$0.00000040	Queue Storage Req	$0.00000040	Cloud Pub/Sub Req	$0.00000034
SNS Request	$0.00000060	Notification Hubs Req	$0.00000060	Cloud Pub/Sub Req	$0.00000051
Route53 Query	$0.00000040	DNS Query	$0.00000040	Cloud DNS Query	$0.00000038
VPC DataProcessed-Bytes	$0.01000000	VNet Data Processed	$0.01000000	VPC Network DataProc	$0.01000000
CloudFront DataTransfer-Out-Bytes	$0.08500000	CDN Data Transfer	$0.08075000	Cloud CDN Data Trans	$0.07650000

Figure 3: Multi Cloud Pricing

A few services from three main cloud service providers are compared in terms of costs in the table shown in Figure 3: AWS, Azure, and GCP. Each row indicates a particular service or feature that these platforms provide. For instance, the EC2 t2.micro instance from AWS costs 0.0126 US Dollars per hour, whereas the B1 Series from Azure costs 0.016 US Dollars, and the f1-micro from Google Cloud Platform costs 0.009 US Dollars. Similar comparisons are conducted for various sizes of compute instances, database services, storage activities, and additional functions like serverless computing and messaging as we move down the table. In particular, each service has its own cost measure; for instance, storage operations are paid every GET, PUT, or COPY request. Each request for Lambda, Functions, and Cloud Functions is charged separately. cost optimization technique relies on the table as a key tool, which enables it to determine which service among the providers is the most affordable and to suggest it to the client based on their application requirements.

5 Implementation

As shown in Figure 1 there are the following stages involved in the implementation of the model which includes Data Generation, Time series forecasting of Client Application, and Predicting the next 4 year's data with better Multi-Cloud Services

5.1 Data Generation

Data Generation involves the following steps as shown in Figure 4

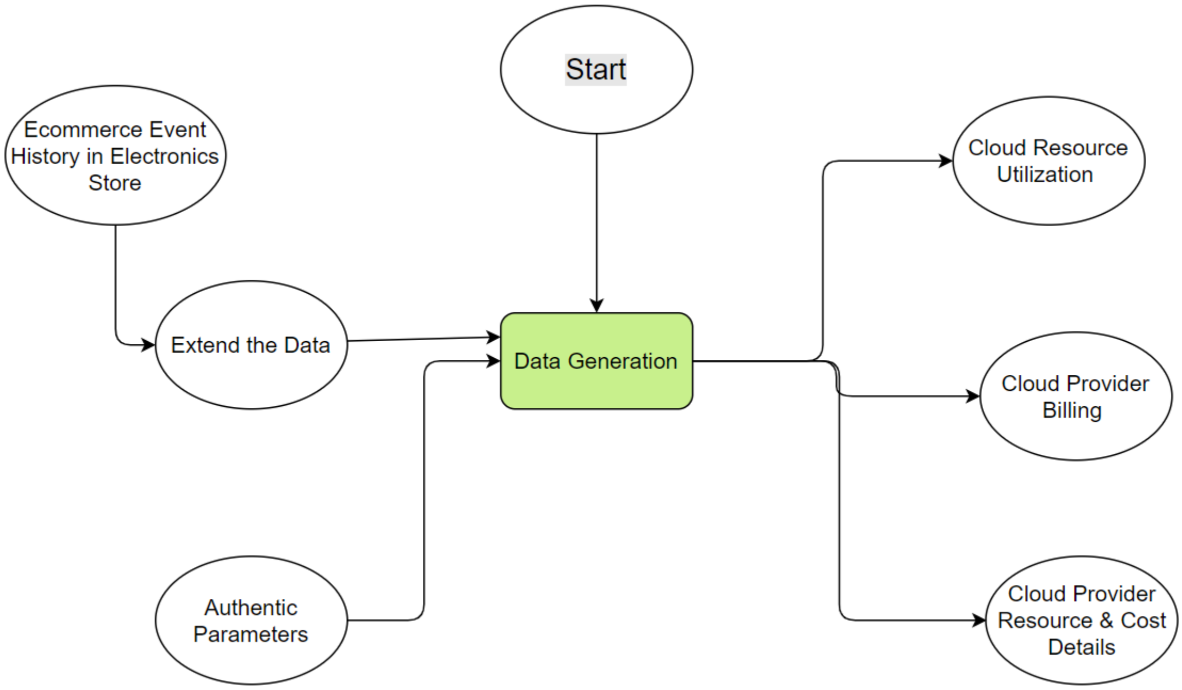

Figure 4: Data Generation

5.1.1 Getting Event Data from Ecommerce Extension

Got data from Kaggel and transform it by increasing the size of the data. The following sections contain the implementation of this feature:

About the Data from Kaggle

The data which is sourced from Kaggle comes from the dataset titled "*eCommerce events history in electronics store* (2021)" as shown in Figure 5, the structure of the dataset. This dataset holds behavior data over 5 months (from October 2019 to February 2020) of users' interactions within a large electronics online store.

The main feature which is considered in the data set is Event Time because this will be the feature we will eventually use to get realistic usage of Client Applications on a Multi-Cloud Platform.

event time: This captures the precise moment (in UTC) when an event took place.
Other feature like event type, cart, remove from cart, product id, category id, category code, brand, price, user id, and user session are also in the file but we are not using them for our research purpose.

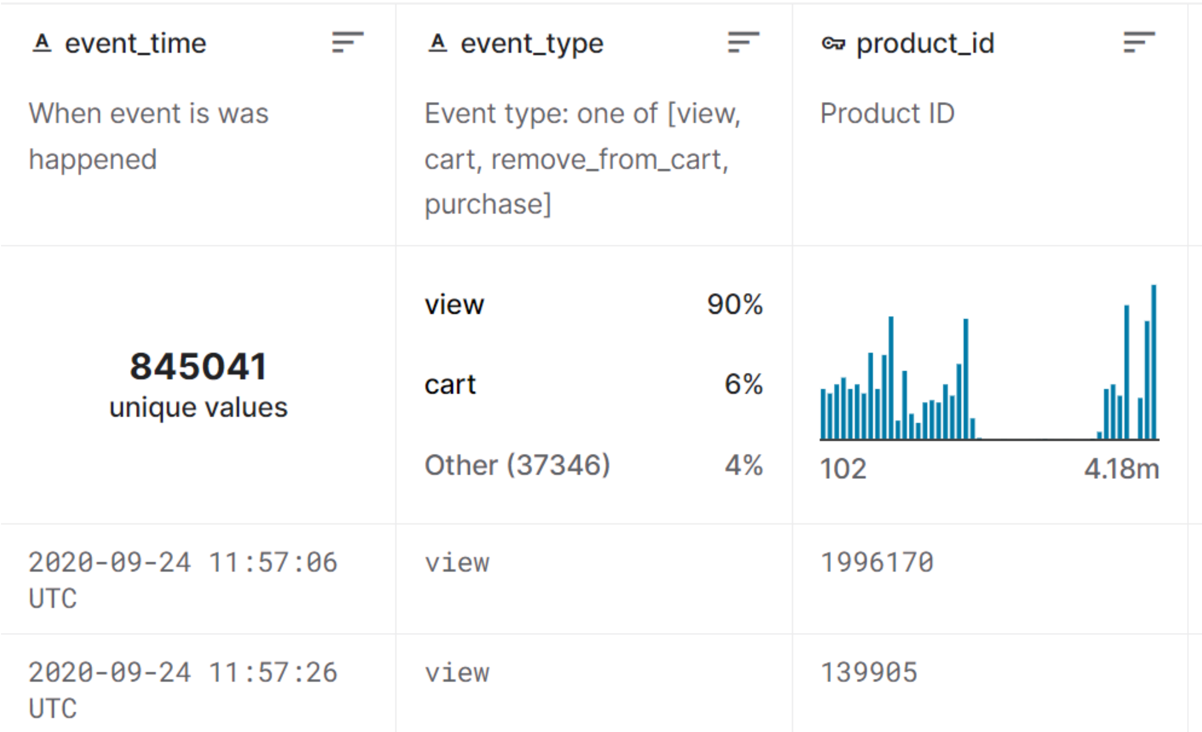

Figure 5: eCommerce events history in electronics store Data

Explanation of the Python Script:

For data transformation and extension Python script is used which undertakes the process of loading, processing, and extending the data:

Loading Data:

The script uses the *Pandas Library* (2023) pandas library to load data from the events.csv file into a dataframe df.

Date Processing:

The 'event time' column is converted into a datetime format to facilitate time-based operations. The dataframe's index is set to 'event time', enabling more straightforward time-based resampling and slicing operations.

Data Extension:

The dataset is extended to cover two years by repeating the 5-month data four times. A loop achieves this repetition. For each iteration, a new dataframe df temp is created which is an exact copy of the original data. The event time (index) of this temporary dataframe is then offset by 5 months multiplied by the iteration number. This offset data frame is then appended to the df list. Once all repetitions are done, all dataframes in the df list are concatenated together to form a two-year data frame df.

Resampling and Event Counting:

The data is resampled on an hourly basis, and the number of actions/events in each hour is counted. This aggregated number is then multiplied by 10 to amplify the events.

Output Generation:

A new dataframe df output is created to store the hourly count of actions/events. This dataframe is then saved to an 'output.csv' file.

About the final output.csv:

The output.csv file generated by your script consists of the aggregated event count based on hourly intervals across the extended two-year period.

The structure would be:

The structure of the output.csv file in shown in Figure 6

	A	B	C
1	event_time	number_of_actions	
2	2020-09-24 11:00:00+00:00	130	
3	2020-09-24 12:00:00+00:00	2700	
4	2020-09-24 13:00:00+00:00	2760	
5	2020-09-24 14:00:00+00:00	2220	
6	2020-09-24 15:00:00+00:00	2400	
7	2020-09-24 16:00:00+00:00	2040	
8	2020-09-24 17:00:00+00:00	2270	
9	2020-09-24 18:00:00+00:00	2050	

Figure 6: output.csv

Event time represents the start time of each hour over the two-year period. number of actions indicates the total number of events/actions that occurred within that hour, multiplied by 10.

This extended and aggregated data can serve multiple purposes like predicting server loads and user behavior patterns over larger time frames, etc.

5.1.2 Multi cloud application (AWS, AZURE, GCP) Date Collection

The primary goal of this study is to compare the pricing policies of the three most well-known cloud service providers, AWS, Azure, and GCP. The prices that are highlighted in the Figure 7 are taken directly from the price listings for 2023 and serve as the core dataset for this study. With a primary focus on compute and database operations, the table strives to include a variety of services from providers. Notably, the costs given under the compute and database categories were acquired directly through the relevant official sources, guaranteeing their validity and applicability to the research timeframe. The costs for other services, such storage, data transport, and application services, are, on the other hand, approximations. These estimates were created using the most recent data available and broad market trends. This table's goal is to provide a crystal-clear, and complete picture of the financial costs associated with picking a certain cloud platform. This information will be crucial in generating insights and conclusions for our model, eventually assisting stakeholders in selecting the best cloud service.

Cloud Services	Cost
B1 Series	$0.01600000
B1ms Series	$0.02800000
B2s Series	$0.05600000
RDS db.t2.micro	$0.01800000
RDS db.t2.small	$0.03600000
RDS db.t2.medium	$0.07300000
S3 GET	$0.00040000
S3 PUT	$0.00050000
S3 COPY	$0.00050000
S3 POST	$0.00050000
S3 LIST	$0.00500000
DynamoDB Write request unit	$0.00065000
DynamoDB Read request unit	$0.00013000
Lambda Request	$0.00000020
SQS Request	$0.00000040
Cloud Pub/Sub Req	$0.00000051
Route53 Query	$0.00000040
VPC Network DataProc	$0.01000000
CloudFront DataTransfer-Out-Bytes	$0.08500000

Figure 7: Client Application Pricing

5.1.3 Using Python Faker Generating Application Resourse Utilization and Billing Data

We start by working with a dataset that records the timestamps of actions. Synthetic data creation techniques are used to enhance this base data and present a plausible cloud use scenario using *Faker Library* (2023). To achieve this, many cloud services must be designed with various consumption patterns. The structure outlines several prices and service kinds, including both request-based and computational services like Azure Compute and RDS.

The algorithm then moves on to summarize the data after producing this comprehensive

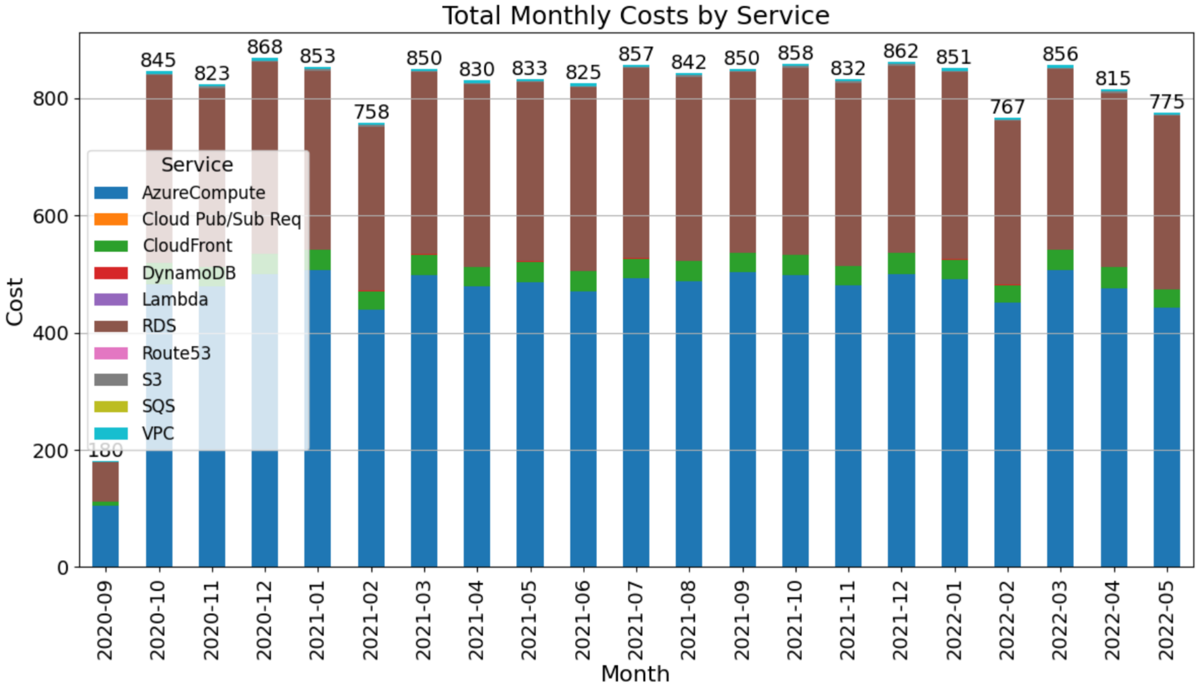

Figure 8: Client Application Billing Data

billing data. The goal is to provide insights that are both detailed, capturing details like CPU use, and comprehensive, showing the total cost of each service. The information is saved in a structured Excel file to make sure it is readily available for additional analysis.

5.2 Time series forecasting of Client Application

Cost reduction is crucial in the world of cloud computing. As businesses continue to move their workloads, data, and apps to the cloud, controlling expenses gets more complex. Although the multi-cloud environment adds another level of complexity, it also presents opportunity to use the greatest features offered by various cloud providers. In this context, predictive analysis might be a crucial tool. The goal of this research is to comprehend how client application expenses will develop over time in a multi-cloud scenario. The forecasting model utilized here is based on fundamental time series forecasting concepts rather than a conventional Machine Learning (ML) model.

5.2.1 Data Acquisition and Preliminaries

Data collection is the first phase in the process. The read excel function of the Python pandas package makes it simple to ingest data by pulling information from the Excel file ”client billing data.xlsx”. The DataFrame's DataFrame has loaded data that shows historical billing information about a client's usage of various cloud services.

5.2.2 Forecast Assumptions

The predictive model makes the assumption that expenses and utilization would rise steadily during the forcasted time period. This model is simplified, thus it might not be able to detect complex patterns that can be detected by ML regression models or conventional time series models.

Two main percentage increase metrics are defined:

cost increase percentage: Represents the anticipated annual rate of increase in costs, which is considered a 1 percent increase per year.

usage increase percentage: Represents the expected yearly growth rate in the usage of various services, which is considered a 20 percent increase per year.

5.2.3 Data Preparation and Forecasting

The core of forecasting is an iterative loop that spans the following four years. **For each year:**

The original data is copied into a temporary DataFrame. The number of the predicted year is added to the year in the day column. Based on the assumed percentage increases, costs and use metrics are modified. Some services like "AzureCompute" and "RDS" see an increase in the number of instances, while others like "S3" and "DynamoDB," among others see a spike in requests. The projected DataFrame is then supplemented with the updated data for the predicted year.

5.2.4 Data Export and Aggregation

day	hour	service	service_type	num_instances	cpu_utilization	cost
2020-09-24	11	AzureCompute	B1 Series	20	0.53	0.32
2020-09-24	11	RDS	db.t2.medium	10	0.59	0.73
2020-09-24	11	S3	POST			0.000065
2020-09-24	11	DynamoDB	Read request unit			0.0000169
2020-09-24	11	Lambda	Request			0.000000026
2020-09-24	11	SQS	Request			0.000000052
2020-09-24	11	Cloud Pub/Sub	Request			6.63E-08
2020-09-24	11	Route53	Query			0.000000052
2020-09-24	11	CloudFront	DataTransfer-Out-Bytes			0.072849325
2020-09-24	11	VPC	DataProcessed-Bytes			0.001622952

Figure 9: 4 year Forecast of Client Application Billing Structure

Following the completion of the forecast, the data is written into an Excel file with the following summaries:

Detail Data: This provides a thorough breakdown of the predicted data as shown in Figure 9. The final structure of the Excel document includes complete data covering about **587,000** rows. These rows provide a four-year predictive projection of resource utilization and billing.

Cost Summary: Aggregated data showing total forecasted costs by service.
Instances Summary: Mean metrics like the number of instances and CPU utilization for services that employ instances.
Requests Summary: The total number of requests aggregated by service.
Data Processed Summary: Summarized data for services processing data.
Data Transfer Summary: Aggregated data transfer metrics for specific services.

These reports give a comprehensive picture of the projected costs for the multi-cloud environment, making it easier to allocate resources and create strategic budgets. Figure 8 displays a bar chart of monthly costs, generated from the collected data.

5.3 Predicting next 4 year data with better Multi Cloud Services

In previous section 4 years client application billing and resource utilization data was predicted without any modification to the selected service but in this phase implementation is concerned with gathering and modifying past billing data. The prediction model is built around the data which is gathered from an Excel spreadsheet which was finalised in Section 5.1.3. This dataset has been updated to match the terminology that has changed quickly with the cloud computing industry. In particular, services formerly known as "AzureCompute" is rephrased as "Multicloud Compute." By adopting a multicloud ecosystem over a single cloud provider strategy client was originally using Azure Compute Services, this rephrasing represents a strategic shift. The groundwork for cost optimization across a wide range of cloud services is laid out in this way.

A carefully planned conversion mapping is the foundation of the cost optimization plan. This map is a dynamic tool intended to redeploy services and resources from their current configurations to more affordable alternatives inside the multicloud environment. It goes beyond simple static transfer of services. Importantly, there are other factors to consider outside the map's static design when deciding whether to transfer service. Decisions are based on dynamic operational parameters like the typical CPU use of a service. This guarantees that, even if cost savings is a top priority, the operational integrity and performance requirements of client applications are not compromised.

The next step is to create a reliable projection for the following four years using the updated data. This predictive methodology goes beyond simple linear cost projections. It includes predetermined annual costs and uses increases that simulate the natural growth and development of an organization's cloud consumption patterns. The approach gives businesses the power to alter their multicloud strategy in advance by producing this thorough foresight. The capacity to make data-driven decisions regarding resource deployments and modifications on their multicloud platforms, as well as more effective budgeting and improved awareness of future financial outlays, are all benefits they may take use of.

After the procedure is complete, the complex fabric of projected data is organized in a database and exported as an Excel worksheet. The reliable arrangement of this final collection guarantees easy accessibility for all parties. The final structure of the Excel

day	hour	service	service_type	num_instances	cpu_utilization	cost
2023-10-01 00:00:00	0	AzureCompute	B1ms Series	24	0.67	0.5656
2023-10-01 00:00:00	0	RDS	db.t2.micro	12	0.92	0.1818
2023-10-01 00:00:00	0	S3	COPY			0.0003232
2023-10-01 00:00:00	0	DynamoDB	Read request unit			0.000084032
2023-10-01 00:00:00	0	Lambda	Request			1.2928E-07
2023-10-01 00:00:00	0	SQS	Request			2.5856E-07
2023-10-01 00:00:00	0	Cloud Pub/Sub Req	Request			3.29664E-07
2023-10-01 00:00:00	0	Route53	Query			2.5856E-07
2023-10-01 00:00:00	0	CloudFront	DataTransfer-Out-Bytes			0.013248952
2023-10-01 00:00:00	0	VPC	DataProcessed-Bytes			0.002846186

Figure 10: 4 year Forecast of Client Application Billing with Cost Optimization Sheet Structure

document includes complete data covering about **587,000** rows, as seen in Figure 10. These rows provide a four-year predictive projection of resource utilization and billing.

6 Evaluation

There are three separate usage scenarios discussed in this section. Three client apps were subjected to predictive analysis, one using Microsoft Azure machines as compute instances, the second using GCP compute instances, and the third using AWS Compute Instances. For each usage scenario, there is one line chart in this section.

6.1 Client Application using Microsoft Azure machines as main Compute instances

Cost prediction for 4 years is shown in Figure 11 in Blue lines having **Grand Total: 41062.98 US Dollars** and after applying cost optimization technique, 4 years prediction is shown with yellow lines with reduced cost in Figure 11 having **Grand Total: 27609.83 US Dollars**.

6.2 Client Application using GCP machines as main Compute instances

Cost prediction for 4 years is shown in Figure 12 in Blue lines having **Grand Total: 39449.85 US Dollars** and after applying cost optimization technique, 4 years prediction is shown with Yellow lines with reduced cost in Figure 12 having **Grand Total: 27317.83 US Dollars**.

6.3 Client Application using AWS machines as main Compute instances

Cost prediction for 4 years is shown in Figure 13 in Blue lines having **Grand Total: 37829.30 US Dollars** and after applying cost optimization technique, 4 years prediction

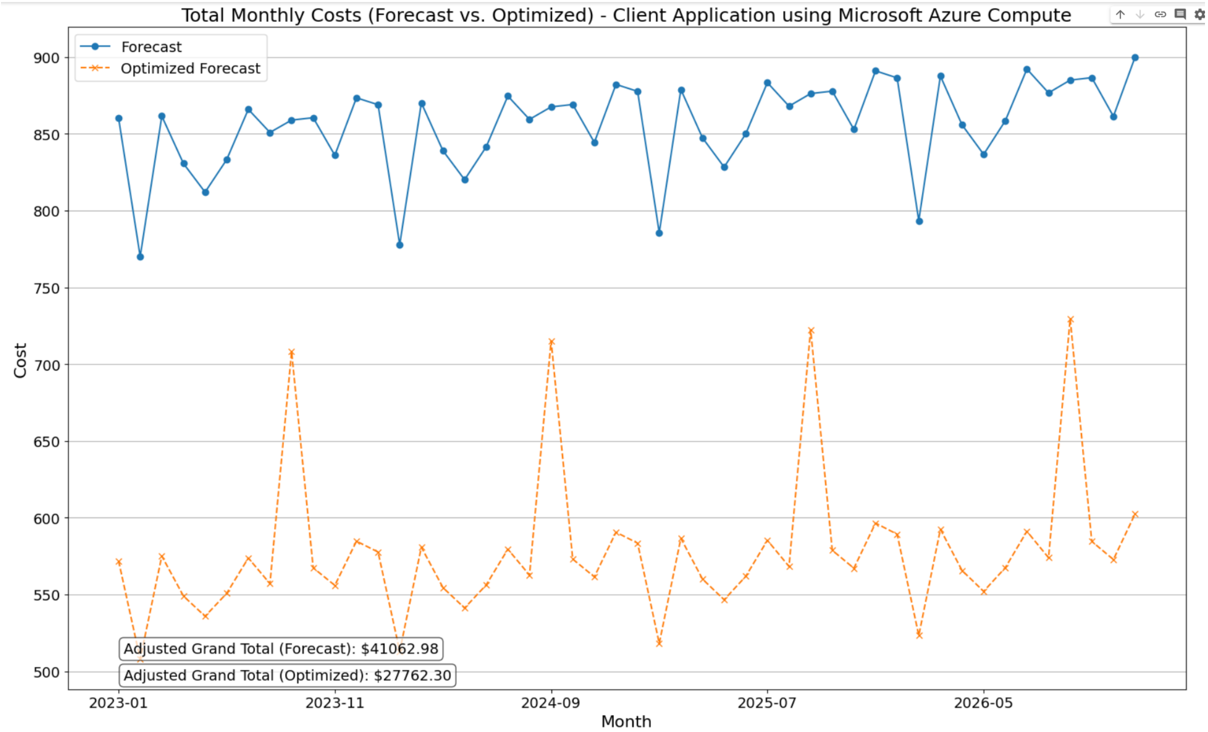

Figure 11: **Scenario 1** - 4 year Forecast of Client Application Billing (Normal vs. Optimized)

is shown with yellow lines with reduced cost in Figure 13 having **Grand Total: 25639.63 US Dollars**.

6.4 Discussion

The huge savings made achievable by the used cost optimization approach were made clear when a detailed analysis of the tests was performed. Figure 11 shows an approximate **33.6%** save after cost optimization model implementation. Figure 12 shows an approximate **30.77%** save after cost optimization model implementation. Figure 13 shows an approximate **32.21%** save after cost optimization model implementation, providing strong support for the model's effectiveness.

When compared to the research revealed during the literature study, these results show that our model's capacity to forecast and minimize costs is on track with, if not better than, many currently used approaches. The experiment does have the potential for improvement. For instance, even if the model's data transformation and service-to-service mapping techniques worked well, the dynamic changes based on operational metrics still have room for improvement. A more detailed analysis of measurements or even taking into account performance factors other than CPU utilization might improve the prediction's accuracy and lead to greater cost savings.

The design might be criticized for being heavily dependent on previous billing data, despite the fact that it is data-driven. The complicated nature of future pricing changes for cloud services or unexpected operational needs may not always be captured by relying

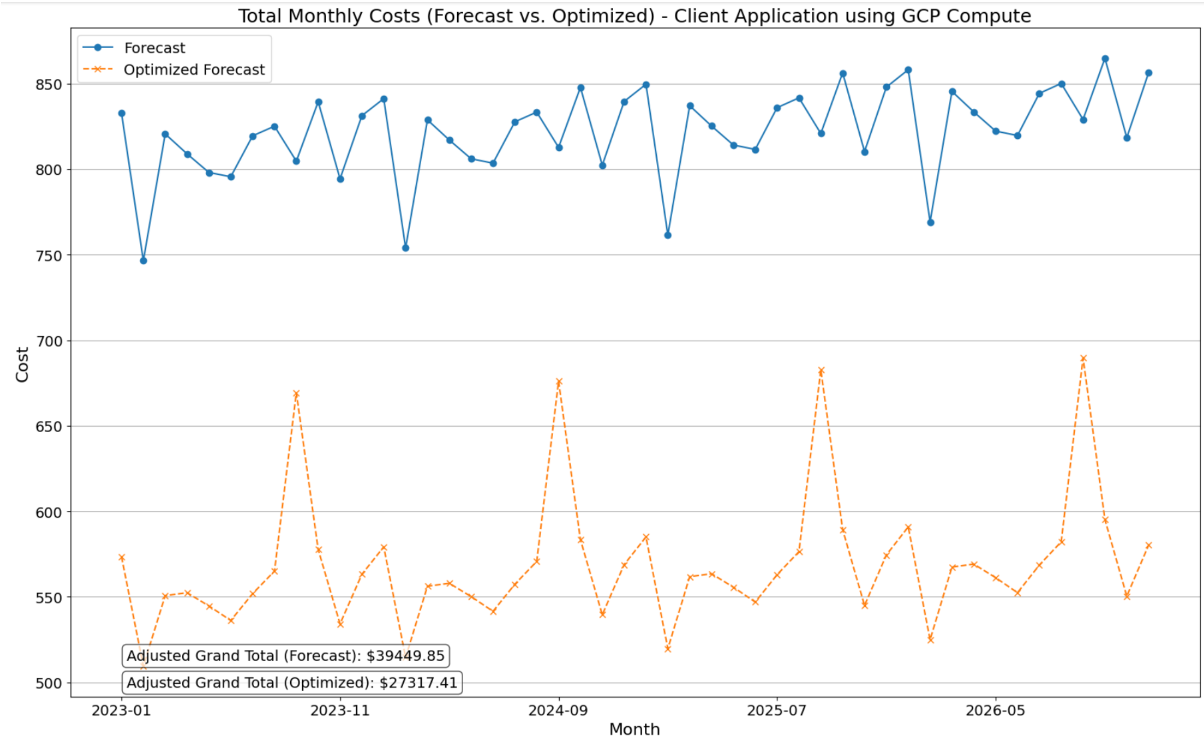

Figure 12: **Scenario 2** - 4 year Forecast of Client Application Billing (Normal vs. Optimized)

mainly on historical data. An accurate forecast may be provided via a hybrid model that combines historical data with real-time monitoring and predictive analysis.

7 Conclusion and Future Work

Deploying multi-cloud systems is becoming more popular in the current digital era as businesses seek the most effective platforms to accomplish their goals. This study examines a predictive analytical model created especially to meet this growing requirement, with a particular focus on the three main cloud service providers, AWS, Azure, and GCP. The research's unique aspect goes beyond only forecasting future costs; it is its sophisticated optimization method. It carefully identifies and utilizes the cloud platforms' most economical services. In order to ensure maximum financial effectiveness, organizations might strategically use their resources rather than simply absorb expenses. The data collected from this research, covering a predicted four-year period, illustrates the model's ability and shows the possibility of significant cost savings for businesses.

There is an opportunity for improvement even though the existing model provides important knowledge about multi-cloud estimations of costs and optimization. To provide a more thorough perspective, future versions could include a wider range of cloud service providers and their services. Additionally, incorporating machine learning techniques might improve the predicted accuracy over time by learning from and adapting to real-time billing data. Investigating automation's involvement in dynamically switching between cloud services depending on price changes and service demand might also be

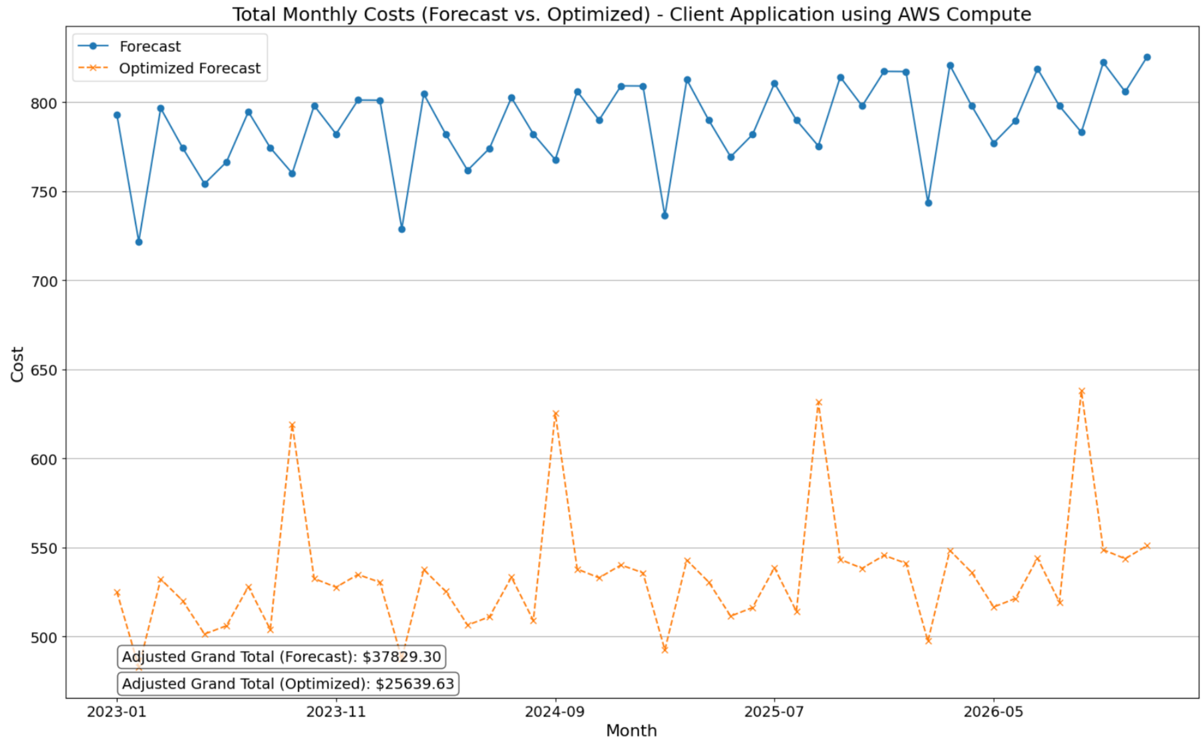

Figure 13: **Scenario 3** - 4 year Forecast of Client Application Billing (Normal vs. Optimized)

helpful. Ultimately, future models may incorporate environmentally friendly measures to ensure both economic and ecological efficiency in multi-cloud systems, taking into account the environmental effect of cloud installations.

References

Addya, S. K., Satpathy, A., Ghosh, B. C., Chakraborty, S., Ghosh, S. K. and Das, S. K. (2021). Comcloud: Virtual machine coalition for multi-tier applications over multi-cloud environments, *IEEE Transactions on Cloud Computing* .

Alonso, J., Orue-Echevarria, L., Casola, V., Torre, A. I., Huarte, M., Osaba, E. and Lobo, J. L. (2023). Understanding the challenges and novel architectural models of multi-cloud native applications–a systematic literature review, *Journal of Cloud Computing* **12**(1): 1–34.

Alshammari, M. M., Alwan, A. A., Nordin, A. and Al-Shaikhli, I. F. (2017). Disaster recovery in single-cloud and multi-cloud environments: Issues and challenges, *2017 4th IEEE International Conference on Engineering Technologies and Applied Sciences (ICETAS)*, pp. 1–7.

Belgacem, A. (2022). Dynamic resource allocation in cloud computing: analysis and taxonomies, *Computing* **104**(3): 681–710.

Belusso, C. L. M., Sawicki, S., Basto-Fernandes, V., Frantz, R. Z. and Roos-Frantz, F. (2018). A proposal of infrastructure-as-a-service providers pricing model using linear regression, *Revista Brasileira de Computação Aplicada* **10**(2): 44–53.

de Carvalho, J. O., Trinta, F., Vieira, D. and Cortes, O. A. C. (2018). Evolutionary solutions for resources management in multiple clouds: State-of-the-art and future directions, *Future Generation Computer Systems* **88**: 284–296.

Di Modica, G., Di Stefano, A., Morana, G. and Tomarchio, O. (2019). On the cost of the management of user applications in a multicloud environment, *2019 7th International Conference on Future Internet of Things and Cloud (FiCloud)*, IEEE, pp. 175–181.

eCommerce events history in electronics store (2021). `https://www.kaggle.com/datasets/mkechinov/ecommerce-events-history-in-electronics-store`.

Faker Library (2023). `https://faker.readthedocs.io/en/master/`.

Georgios, C., Evangelia, F., Christos, M. and Maria, N. (2021a). Exploring cost-efficient bundling in a multi-cloud environment, *Simulation Modelling Practice and Theory* **111**: 102338.
URL: *https://www.sciencedirect.com/science/article/pii/S1569190X2100054X*

Georgios, C., Evangelia, F., Christos, M. and Maria, N. (2021b). Exploring cost-efficient bundling in a multi-cloud environment, *Simulation Modelling Practice and Theory* **111**: 102338.

Heilig, L., Lalla-Ruiz, E. and Voß, S. (2016). A cloud brokerage approach for solving the resource management problem in multi-cloud environments, *Computers & Industrial Engineering* **95**: 16–26.

Heilig, L., Lalla-Ruiz, E. and Voß, S. (2020). Modeling and solving cloud service purchasing in multi-cloud environments, *Expert Systems with Applications* **147**: 113165.
URL: *https://www.sciencedirect.com/science/article/pii/S0957417419308826*

Horn, G., Skrzypek, P., Materka, K. and Przeździk, T. (2019). Cost benefits of multi-cloud deployment of dynamic computational intelligence applications, *Workshops of the International Conference on Advanced Information Networking and Applications*, Springer, pp. 1041–1054.

Kumar, M., Sharma, S. C., Goel, A. and Singh, S. P. (2019). A comprehensive survey for scheduling techniques in cloud computing, *Journal of Network and Computer Applications* **143**: 1–33.

Megahed, A., Nazeem, A., Yin, P., Tata, S., Nezhad, H. R. M. and Nakamura, T. (2019). Optimizing cloud solutioning design, *Future Generation Computer Systems* **91**: 86–95.

Mishra, S. K., Mishra, S., Alsayat, A., Jhanjhi, N., Humayun, M., Sahoo, K. S. and Luhach, A. K. (2020). Energy-aware task allocation for multi-cloud networks, *IEEE Access* **8**: 178825–178834.

Osypanka, P. and Nawrocki, P. (2022). Resource usage cost optimization in cloud computing using machine learning, *IEEE Transactions on Cloud Computing* **10**(3): 2079–2089.

Pandas Library (2023). `https://pandas.pydata.org/`.

Pandey, A., Calyam, P., Lyu, Z., Wang, S., Chemodanov, D. and Joshi, T. (2022). Knowledge-engineered multi-cloud resource brokering for application workflow optimization, *IEEE Transactions on Network and Service Management* pp. 1–1.

Ramesh, A., Pradhan, V. and Lamkuche, H. (2021). Understanding and analysing resource utilization, costing strategies and pricing models in cloud computing, *Journal of Physics: Conference Series* **1964**(4): 042049.
URL: *https://dx.doi.org/10.1088/1742-6596/1964/4/042049*

Rossi, C. (2017). Rapid release at massive scale, *Web site: https://engineering. fb. com/web/rapid-release-at-massive-scale* .

Sangeetha, S. B., Sabitha, R., Dhiyanesh, B., Kiruthiga, G., Yuvaraj, N. and Raja, R. A. (2022). Resource management framework using deep neural networks in multi-cloud environment, *Operationalizing Multi-Cloud Environments: Technologies, Tools and Use Cases* pp. 89–104.

Shi, T., Ma, H., Chen, G. and Hartmann, S. (2021). Cost-effective web application replication and deployment in multi-cloud environment, *IEEE Transactions on Parallel and Distributed Systems* **33**(8): 1982–1995.

Singh, P., Gupta, P. and Jyoti, K. (2019). Tasm: technocrat arima and svr model for workload prediction of web applications in cloud, *Cluster Computing* **22**(2): 619–633.

Tang, X. (2022). Reliability-aware cost-efficient scientific workflows scheduling strategy on multi-cloud systems, *IEEE Transactions on Cloud Computing* **10**(4): 2909–2919.

Zhang, P., Zhou, M. and Wang, X. (2020). An intelligent optimization method for optimal virtual machine allocation in cloud data centers, *IEEE Transactions on Automation Science and Engineering* **17**(4): 1725–1735.