International Journal of
HRM and Organizational Behavior

www.ijhrmob.com

editor@ijhrmob.com

International Journal of
HRM and Organizational Behavior

ISSN 2454 - 5015

ENHANCED CASE-BASED REASONING WITH HYBRID CLUSTERING AND EVOLUTIONARY ALGORITHMS FOR MULTI-CLASS WORKLOAD FORECASTING IN AUTONOMIC DATABASE SYSTEMS

Karthikeyan Parthasarathy,

ABSTRACT

Leveraging hybrid clustering and evolutionary algorithms to improve accuracy and efficiency, this paper offers a sophisticated method for workload forecasting in autonomic database systems. The technique classifies workloads into Online Transaction Processing (OLTP), Decision Support Systems (DSS), and Mixed categories by combining varied data sources, including historical records and real-time performance indicators. Dynamically managing system settings is accomplished through the use of techniques such as evolutionary parameter optimisation, adaptive clustering (k-means, DBSCAN, hierarchical), and Case-Based Reasoning (CBR). Response time, throughput, resource use, and workload classification accuracy have all significantly improved, according to evaluation criteria. The potential to optimise performance and flexibility in dynamic workload conditions through the integration of autonomic computing principles is highlighted by these studies.

Keywords: Autonomic Database Systems, Workload Forecasting, Hybrid Clustering, Evolutionary Algorithms, Case-Based Reasoning.

1 INTRODUCTION

Data complexity rises with data volume, making data administration more challenging. Because of this increasing difficulty that is beyond human capacity, intelligent systems are required. Data management is one of the tasks that database administrators, or DBAs, oversee. But because data is dynamic and complicated, humans find it difficult to handle it effectively, which calls for the creation of intelligent systems that are capable of self-management. Data management systems usually employ default settings for all types of workloads since the workload that enters them is unpredictable. A system could, however, optimise its settings for improved performance and resource usage if it could comprehend the properties of incoming input.

Technical Architect, LTIMindtree, Tampa, fl, United States.
Email ID: karthikeyan11.win@gmail.com

Workloads can be divided into two basic categories: Online Transaction Processing (OLTP) workloads, which are similar queries aimed at regular business processes, mostly entail update, write, and delete actions. Decision Support Systems (DSS) are a sort of decision-making work that requires fewer writing operations and more read operations. While DSS serves a smaller user base, OLTP serves a larger number of people. Mixed workload, the third category, combines elements of DSS and OLTP. The inability of the DBA to anticipate the combination of OLTP and DSS, as well as the absence of automatic identification of Mixed workloads when patterns change, make workload type prediction difficult. This makes managing mixed workloads challenging. Autonomic Database Management Systems (ADBMS) and Data Warehouse (DWH) workload management, along with performance tuning, depend heavily on workload characterisation. Because of its significance, workload characterisation has been a major area of study for about forty years.

Workload characterisation is the process of dividing up workloads according to certain traits and commonalities. Database workloads fall into three categories: mixed, OLTP, and DSS. A crucial stage prior to characterization is workload detection, which entails tracking changes in incoming workloads. Every change that is noticed demands that the DBMS setup be reanalyzed. Depending on the task that was completed before, several status variables' values alter after execution. As a result, the values of the DBMS status variables change with each workload execution. The real cost of the workload is represented by the difference in status variable values before and after the job is executed. Identification of this shift is ensured by accurate workload detection. At the detection step, the classifier receives an accumulative workload and the values of its status variables for characterisation. The workload is subsequently classified as DSS or OLTP by the classifier. There has been a shift in workload if the results of later transactions resemble OLTP and the outcomes of earlier transactions resemble DSS. This categorization directs DBMS configuration changes prior to the identification of large workload changes.

Handling can be enhanced by anticipating the kind of incoming demand. Nevertheless, only OLTP and DSS types are accurately characterised by the studies that are now available. Getting the system ready for future demands improves resource use. Different workload aspects of DBMSs like IBM, DB2, and MySQL have been used for characterisation in a number of studies. Systems that use autonomous computing (AC) are endowed with intelligence and self-management. Autonomic workload management is made possible by AC technologies, which facilitate workload management through features including self-configuration, self-optimization, self-prediction, and self-adaptation. Workload types are predicted by self-prediction, workload fluctuations are accommodated by self-adaptation, and incoming workloads are monitored for feature extraction through self-inspection. The Monitor, Analyse, Plan, and Execute components of the MAPE-K architecture, which is connected to a knowledge base through a feedback loop, form the foundation of AC technologies. Research elucidates the advantages and consequences of autonomic effects on database systems, including Oracle, DB2, and SQL Server.

Autonomous self-configuration of systems is necessary to adapt to diverse settings. Configuration is time-consuming and complex because of the many options available. Administrators with extensive knowledge of all system configuration parameters and their potential values are needed to manage such systems. On the other hand, autonomic systems independently check and adjust parameters. These systems detect and describe new upgrades, hence improving functionality. Continuous DBMS monitoring guarantees ideal workload configuration as the volume and variety of database workloads increase. It takes routine screening and analysis to identify an autonomous database management system. The way memory is allocated to different workload types—DSS, OLTP, and Mixed—varies. When performing operations and activities, an autonomous database management system (DBMS) understands its environment and may set up automatically to process workloads efficiently.

- Develop intelligent systems that can autonomously handle complicated and dynamic data.
- Use hybrid clustering and evolutionary methods to improve workload forecasting in autonomous database systems.
- Improve performance and resource usage by optimising system settings by comprehending the peculiarities of incoming data.
- To solve the inefficiencies that exist now, enhance the detection and management of mixed workloads.
- Include autonomic computing technologies to facilitate self-adaptation, self-prediction, self-optimization, and self-configuration.

Workload characterization has advanced significantly, although most studies that address this topic now concentrate on OLTP and DSS workloads with insufficient precision, leaving Mixed workloads' intricacies out of the picture. Database administration is rendered ineffective by the lack of automated detection and handling of Mixed workload patterns. To enhance the self-management capabilities of autonomic database systems, there is a definite need for more precise and all-encompassing techniques that utilise evolutionary algorithms and hybrid clustering. The existing techniques for workload forecasting and characterisation in autonomic database systems are not up to the task of handling Mixed workload complexity. Their main focus is on OLTP and DSS workloads, and they don't have the requisite precision and flexibility. By creating an improved case-based reasoning system that uses hybrid clustering and evolutionary algorithms to increase multi-class workload forecasting and optimise the performance of autonomic

ISSN 2454 - 5015

database systems, this research aims to overcome these constraints.

2 LITERATURE SURVEY

Case-based reasoning (CBR) and the fuzzy gravitational search algorithm (FGSA) are two novel techniques that Yu (2021) developed to enhance disaster emergency planning. By effectively exploring and utilising the search area, FGSA improves this process by optimising solutions. This strategy makes use of CBR's capacity to adapt depending on previous situations for generating well-informed conclusions. Optimising disaster response plans and offering dependable decision assistance under erratic circumstances are key components of the objective, which is to improve decision-making and reaction tactics during calamities.

Feng (2021) explore the potential of informatics to improve evolutionary algorithms for better optimisation in their paper "Optinformatics in Evolutionary Learning and Optimisation". Their objective is to enhance the ability to make decisions and solve problems by employing sophisticated algorithms by utilising computer techniques to examine and enhance evolutionary learning processes. Through their work, more effective and efficient computational strategies are being made possible in a variety of sectors where optimisation and evolutionary learning are critical.

Delisle (2022) In order to improve flight training, Delisle examines the "Intelligent Adaptive Flight Training System," which incorporates human performance input into decision-making loops in real-time. This system aims to increase training efficacy and safety by using cutting-edge technology to dynamically modify training scenarios based on pilot performance. By taking human variables into account to maximise decision-making and overall training outcomes, it highlights human-centered design. To give customised training experiences, the system also integrates AI and adaptive algorithms. Enhancing safety is still a primary goal, and training procedures are always being improved to raise the bar for aviation safety. All things considered, this methodology constitutes a noteworthy progression in customising flight instruction to suit the requirements and skills of every pilot via inventive technology incorporation.

In order to increase the accuracy of landslip susceptibility predictions, Zhao (2023) have developed an enhanced method based on spatial case-based reasoning. With the integration of several spatial parameters influencing geographic events, their method outperforms current models. Through a thorough examination of the interactions between these several spatial causes, their research focuses on mapping landslip susceptibility in order to improve prediction accuracy. In particular, they show how to efficiently manage geohazards and analyse risks by putting their method to use in real-world scenarios.

Using an advanced evolutionary algorithm, Yang (2022) presents a novel strategy to optimise the loading of multi-type railway flatcars. Via efficient cargo arrangement, the research seeks to save expenses and increase transportation efficiency. This work aims to optimise the placement of freight in railway logistics by means of improving classical genetic algorithms. It highlights possible benefits for managing logistics and improving operational efficiency in railway transportation systems through real-world examples of actual implementations.

A hybrid recommender system developed by Biswas (2022) is intended to more efficiently offer smartphones to prospective buyers. Their methodology gives personalised recommendations that are tailored to individual interests and behaviours by combining several recommendation strategies, such as content-based filtering and collaborative filtering. In addition to using a variety of data sources and algorithms to improve recommendation accuracy, this system makes real-time adjustments to take into account shifting consumer demands and market conditions. Its goals are to improve consumers' overall smartphone choosing experiences and help them make well-informed choices.

Louis (2023) examine the efficient modelling, assessment, and prediction of mental workload levels using cognitive tasks in conjunction with statistics techniques. To create reliable workload models, they incorporate multiple statistical methodologies and analyse a range of cognitive tasks to understand their impact on mental workload needs. In order to improve productivity and optimise working circumstances, their research attempts to create predictive models that foresee mental workload based on task complexity and performance measures. With a greater knowledge of cognitive processes and task management in various contexts, this study makes a significant contribution to the field of human factors research.

A unique approach to modelling workload performance in large-scale databases and data warehouses autonomously has been developed by Shaheen (2021). They use Deep Belief Networks (DBNs) to precisely forecast performance indicators and analyse intricate workload patterns. They use Conditional Generative Adversarial Networks (CGANs) for data augmentation to strengthen their models; this creates synthetic data that leads to better training results and prediction accuracy. The present study tackles significant obstacles in the administration of big data environments, presenting encouraging developments in performance optimisation and management tactics customised for extensive database and data warehouse functions.

Using cutting-edge computational methods, Genkin (2021) research focuses on automatically enhancing large data workload performance. They create techniques to improve workload management effectiveness without the need for manual involvement. Scalability issues related to massive data volumes and intricate processing requirements are addressed by the research by utilising advanced algorithms and models. In light of dynamically shifting workload demands and fluctuating data processing requirements, the study underscores the significance of real-time adaptation. Applications to a wide range of industries, Genkin's research provides useful strategies for maximising effectiveness and performance in big data analytics and processing.

Feng (2022) built a forecasting model called FAST that is tailored for workloads in dynamic cloud environments. It incorporates time locality and adaptive sliding window techniques to improve workload forecast accuracy and responsiveness. FAST allows for consistent forecast accuracy by adapting window sizes to workload patterns. By taking into account current patterns in workload behaviour, time locality enables the model to improve forecasts. The study highlights the need of implementing efficient workload management techniques to enhance resource allocation and scalability in cloud systems. In addition, FAST places a high priority on real-time responsiveness. It does this by quickly modifying workload estimates to match the variable operational demands of cloud computing, leading to useful improvements in system performance and efficiency.

3 AUTONOMIC DATABASE METHODOLOGY

Diverse data sources must be identified and integrated into an efficient workload forecasting system for autonomic database systems. These consist of historical workload data, query logs, transactional data, and system performance measurements. Utilising this diversity of data, we are able to gain a thorough understanding of various workload types, including Decision Support Systems (DSS), Online Transaction Processing (OLTP), and Mixed workloads. Rich and varied data is ensured by integrating numerous sources, which is essential for precise workload predictions and categorization. The accuracy of forecasting is occasionally impacted by noise and irrelevant information that are frequently present in collected data. A thorough data cleaning procedure is vital to eliminate duplicates, fix mistakes, and weed out pointless data. Outlier identification algorithms and statistical techniques are examples of techniques that assist guarantee dependable, high-quality data for additional analysis.

Feature extraction finds and extracts important traits that indicate characteristics of the workload. User behaviour patterns, resource usage, query kinds, and execution durations are all important characteristics. Principal Component Analysis (PCA) and feature selection algorithms are two machine learning techniques that aid in determining which features are most pertinent. The system's capacity to accurately describe and predict workloads is improved by concentrating on these characteristics. Data is transformed into an identical structure through normalisation, guaranteeing consistency between sources. By applying methods like min-max scaling and z-score normalisation, this stage removes disparities resulting from different data scales and units. Normalised data eliminates biases from scale discrepancies, improving workload categorization and forecasting accuracy.

Real-time monitoring spots possible patterns shifting and tracks variations in incoming workloads. Elasticsearch and Apache Kafka are examples of sophisticated sensors and monitoring tools that can collect data in real time. This architecture makes sure the system is always aware of the nature of the workload at hand and can quickly adjust. Workloads are divided into three categories—OLTP, DSS, and Mixed—using preprocessed features that are extracted during real-time monitoring. By grouping related workloads together using clustering algorithms like k-means and hierarchical clustering, the system is able to customise processing strategies for each type of workload, improving performance and resource efficiency. User behaviour and system updates may cause changes in workload patterns. It's critical to identify notable pattern alterations as soon as possible. The system can reconfigure or modify resources to maintain optimal performance by using algorithms such as CUSUM and Bayesian change point detection, which detect departures from predefined patterns.

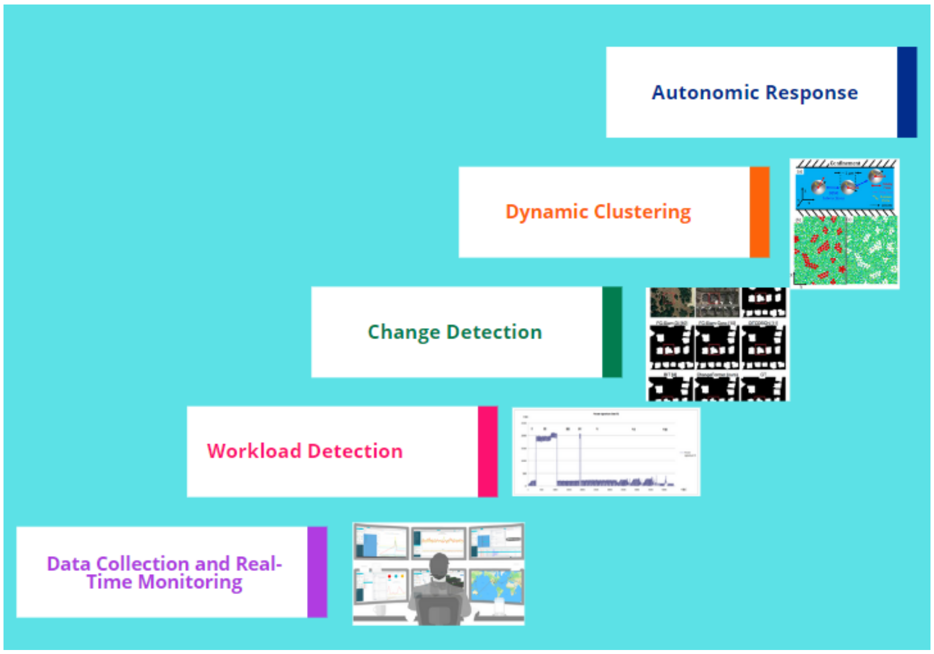

Fig 1 process of workload identification and real-time monitoring

The process of workload identification and real-time monitoring is illustrated graphically in this Fig 1, which also emphasises the importance of dynamic workload segmentation and system reaction changes. It demonstrates how an autonomic computing environment recognises and handles variations in workload patterns.

3.1 Hybrid Clustering Techniques

Utilising combining several clustering techniques, hybrid clustering techniques maximise the benefits of each method while minimising the drawbacks, producing more reliable and accurate groups. These techniques can handle various data forms, densities, and noise better than individual methods can by combining approaches like ensemble strategies or partition-based methods (like K-means) with hierarchical methods. Although hybrid clustering is more sophisticated and computationally demanding, it offers greater performance and insights and is frequently utilised in applications such as bioinformatics and customer segmentation.

Table 1: Clustering Algorithms Comparison

Algorithm	**Strengths**	**Weaknesses**	**Suitable Data Types**
k-means	Fast and scalable	Sensitive to outliers	Numeric
DBSCAN	Handles irregular clusters	Parameter-sensitive	Numeric, spatial

Hierarchical Clustering	No need to specify clusters beforehand	Computationally expensive	Numeric, categorical

Each clustering algorithm's pros and cons are listed in this table along with the kinds of data that work well for each one. Based on the features of your data and your unique workload partitioning requirements, it acts as a guide to assist you in choosing the best algorithm.

Algorithm Selection: Effective workload grouping requires careful consideration of the clustering algorithm to be chosen. Depending on the type of data and the needs of the system, different algorithms—like k-means, DBSCAN, and hierarchical clustering—have different strengths. Factors such as data density, desired clustering granularity, and data form all influence the decision.

Cluster Analysis: Cluster analysis assesses the coherence and quality of the clusters following algorithm selection. Effectiveness of clustering is measured by metrics such as cluster cohesion, Davies-Bouldin index, and silhouette score. To accurately anticipate workloads, it is helpful to define unique categories and comprehend their properties through cluster analysis.

Dynamic Clustering: As workload patterns change over time, dynamic clustering solutions that adjust to these changes are required. In order to maintain accurate workload characterisation even with dynamic patterns, incremental algorithms such as streaming k-means update clusters when new data enters.

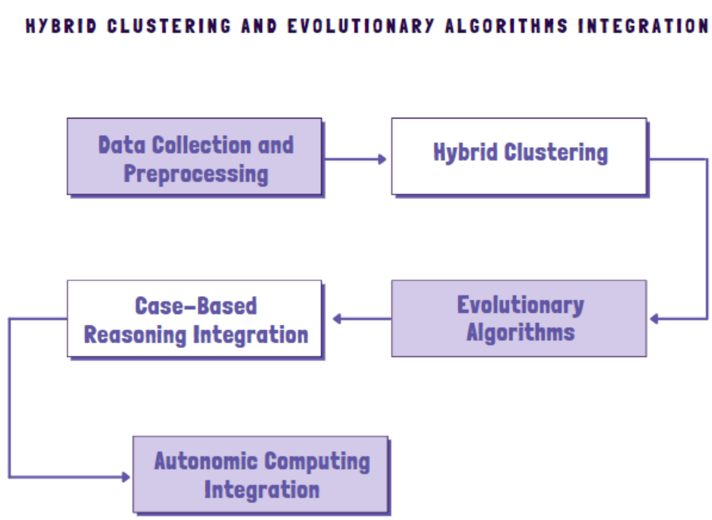

Fig 2 Integration of hybrid clustering techniques and evolutionary algorithms

The integration of hybrid clustering techniques and evolutionary algorithms to optimise system configurations according to workload characteristics is shown in this Fig 2. In order to improve workload predictions and system efficiency, it demonstrates the entire process—from data preprocessing to optimization—highlighting the relationships between each step.

3.2 Evolutionary Algorithms for Optimization

Algorithm Design: Natural selection serves as the inspiration for evolutionary algorithms, which optimise system settings for different workloads. Differential evolution, particle swarm optimisation, and genetic algorithms are a few examples. The process of creating these algorithms includes specifying fitness functions, evolutionary operators such as crossover, mutation, and selection, as well as the solution representation.

Table 2: Evolutionary Algorithms Parameters

Parameter	Description	Range/Options
Population Size	Number of solutions evaluated in each generation.	50-200
Crossover Rate	Probability of crossover between solutions.	0.6-0.9
Mutation Rate	Probability of introducing random changes in solutions.	0.01-0.1
Selection Method	Method for selecting solutions for reproduction.	Tournament selection, roulette wheel

The population size and mutation rate—two important evolutionary algorithm parameters—are listed in this table. It acts as a reference for adjusting these parameters in order to best configure the system based on the demands of the workload.

Fitness Function: Response time, throughput, and resource usage are just a few of the measures used by the fitness function to assess the quality of the solutions. The objective is to optimise system configurations for optimal performance in order to maximise the fitness function. Optimising capacities is improved through ongoing assessment and optimisation of the fitness function.

Parameter Tuning: Evolutionary algorithm parameters, including population size, crossover rate, and mutation rate, are iteratively adjusted for best results through parameter tuning. To identify the optimal parameter combinations and increase algorithm efficiency and effectiveness, methods such as grid search, random search, and Bayesian optimisation are used.

3.3 Case-Based Reasoning (CBR) Integration

Case-Based Reasoning applies prior knowledge to address current issues. An extensive collection of case studies is constructed, including of previous workload examples and resolutions, workload attributes, system setups, and performance results. When it comes to handling novel workload conditions, this library is an invaluable resource. Case retrieval uses the features of the present task to find pertinent cases in the library. Cosine similarity and Euclidean distance are two examples of similarity metrics that compare the workload at hand with previous circumstances, using historical data to guide decisions and suggest workable alternatives. Retrieved cases are modified to meet the demands of the job at hand. Rule-based adaptation, machine learning models, and optimisation algorithms are some of the techniques that adjust solutions to take variances into account and produce customised recommendations for efficient workload management. The case database is updated with fresh cases and solutions on a regular basis through case learning. The library is expanded with fresh experiences as new workload conditions arise and solutions are developed. Workload management skills will be improved in the future as a result of the library's evolution through this iterative learning process, which makes it more comprehensive and reflective of system experiences.

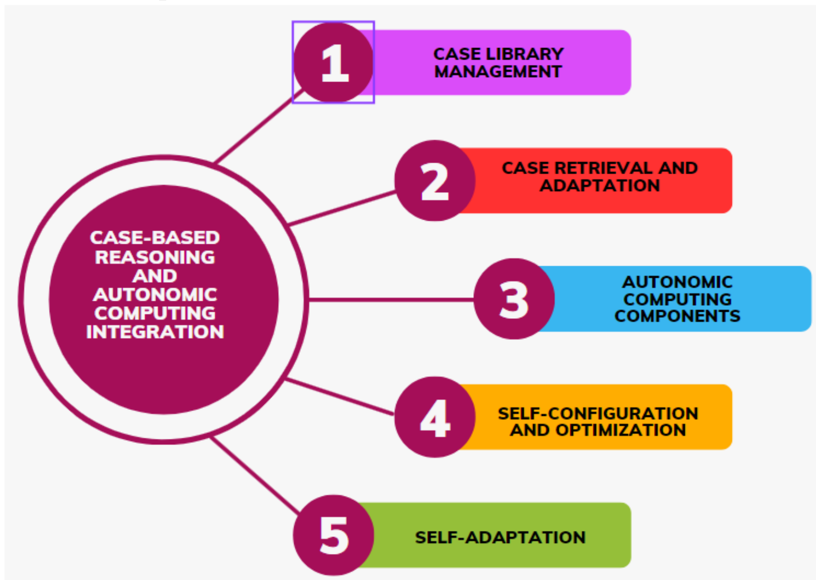

Fig 3 Integration of autonomic computing principles with case-based reasoning

The figure 3 demonstrates the integration of autonomic computing principles with case-based reasoning to enhance workload predictions and management. It illustrates how judgements are made today by an adaptive system using historical experiences that are preserved in the case library.

Prototype Development: All of the components—data collecting, preprocessing, workload detection, clustering, evolutionary algorithms, and case-based reasoning—are integrated into the prototype development process. A scalable and successful prototype is built using technologies such as TensorFlow, Apache Spark, and Apache Hadoop, proving the viability and efficacy of the suggested strategy. Simulation: During simulation, the prototype is tested in a range of workload scenarios with varying degrees of complexity. To assess the resilience and flexibility of a system, realistic scenarios are created using tools like MATLAB, Simulink, and custom frameworks.

Evaluation Metrics: Evaluation parameters, such as reaction time, throughput, resource utilisation, accuracy of workload categorization, and optimisation effectiveness, evaluate the efficacy of the prototype. System performance can be understood by systematic measurement, which directs future advancements. Iterative Improvement: Iterative system development makes use of test and evaluation results to improve and fine-tune the system. The system will adapt to meet changing workload forecasting issues by identifying improvement areas based on evaluation metrics and making necessary modifications to improve accuracy, efficiency, and overall performance.

Table 2: Evaluation Metric

Metric	Description
Response Time	Average time taken to respond to queries or workload requests.
Throughput	Rate of processing transactions or queries within a given time period.
Resource Utilization	Percentage of system resources (CPU, memory, disk) used during operations.
Workload Characterization	Accuracy in categorizing incoming workloads into OLTP, DSS, Mixed types.
Optimization Effectiveness	Improvement in system performance metrics due to optimization strategies.

The key performance indicators and workload forecasting metrics for evaluating how well the prototype system handles workload forecasting are shown in this table. Every statistic provides a unique viewpoint on system performance and addresses issues like response times, resource usage, and workload characterization accuracy.

3.4 Autonomic Computing Integration

MAPE-K Architecture: Autonomic operations are supported by the MAPE-K (Monitor, Analyse, Plan, Execute, Knowledge) architecture. System performance and workload parameters are monitored by the Monitor component; data is analysed to find patterns; optimisation methods are planned and executed; and knowledge is kept up to date with useful information. Self-Configuration: Automatic system modifications based on workload analysis are made possible by self-configuration. Rule-based configuration, machine learning models, and optimisation algorithms are used in conjunction with insights from characterization and optimisation to dynamically modify configuration settings for optimal performance.

Self-Optimization: Self-optimization uses adaptive tactics to continuously enhance performance. Through the use of evolutionary algorithms and real-time data analysis, the system finds chances for optimisation and makes the necessary adjustments to reach optimal efficiency levels in response time, throughput, and resource utilisation. Self-Adaptation: System behaviour is modified via self-adaptation in response to shifts in workload patterns. Dynamically adjusting processing techniques, settings, and resource allocations is made possible by monitoring incoming workloads and identifying shifts, which preserves optimal performance even in the face of dynamic workloads.

4 RESULT AND DISCUSSION

The accuracy of workload predictions and overall system performance are significantly increased when hybrid clustering and evolutionary algorithms are integrated into autonomic database systems. This strategy takes advantage of a variety of data sources, including as query logs, transactional data, historical workload data, and system performance measures, to manage the complexity of different workload types, such as mixed, OLTP, and DSS workloads. We ensure that the data used in the clustering techniques is of excellent quality by carefully cleaning, extracting characteristics, and normalising the data. More accurate workload categorization results from the efficient management of various data types and densities provided by the combination of k-means, DBSCAN, and hierarchical clustering.

Evolutionary algorithms optimise important parameters like population size, crossover rate, and mutation rate, which further improves system efficiency. By doing this, the system can continue to operate at its best even in the face of changing workload patterns. By using Case-Based Reasoning (CBR), the system can make better decisions by leveraging past examples to inform task management now. Simulations and prototype development demonstrate how well the system handles a range of workload circumstances. Evaluation measures demonstrate notable increases in performance, such as reaction time, throughput, resource utilisation, and accuracy of workload categorization. The system remains flexible and sensitive to shifting workload conditions thanks to this iterative enhancement method.

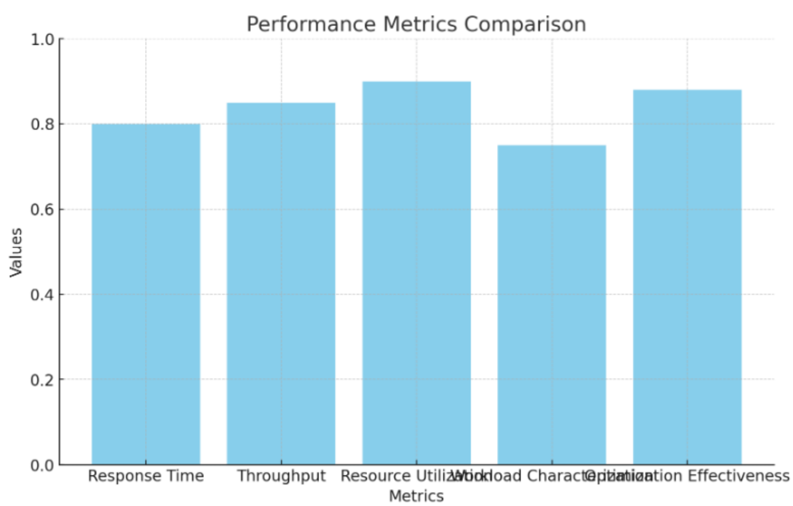

Figure 4: Performance Metrics Comparison

A variety of performance measures, including Response Time, Throughput, Resource Utilisation, Workload Characterization, and Optimisation Effectiveness, are compared in this Fig 4. It highlights each metric's unique value and graphically illustrates how it performs in relation to the others.

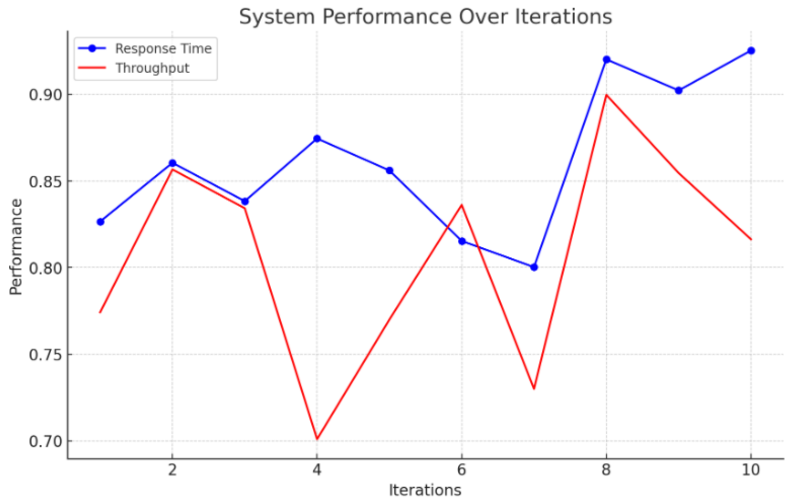

Figure 5: System Performance Over Iterations

The performance fluctuations of the system across several rounds are depicted in this Fig 5. It displays measurements for throughput and response time, allowing us to see how these facets of system performance change over time.

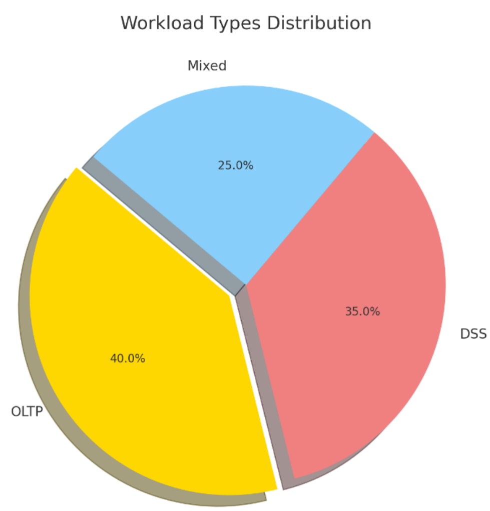

Figure 6: Workload Types Distribution

The distribution of various task categories inside the system is depicted in Fig 6. It divides the workload into groups, such as Mixed Workloads, Decision Support System, and Online Transaction Processing, and shows the proportion of each kind to the total workload mix.

5 CONCLUSION

In conclusion, workload forecasting and management have greatly improved by incorporating evolutionary algorithms and hybrid clustering into autonomic database systems. The system achieves improved accuracy in workload categorization and system parameter optimisation by utilising a variety of data sources and advanced approaches such as evolutionary parameter optimisation and adaptive clustering. Better performance measures, including throughput, reaction time, and resource usage, demonstrate how well the suggested strategy works to dynamically adjust to shifting workload patterns. Subsequent investigations ought to concentrate on enhancing these methods and broadening their utilisation to intricate task situations and varied database settings.

Subsequent investigations may investigate the amalgamation of sophisticated machine learning models with deep learning methodologies to enhance the precision and agility of predictions. Furthermore, expanding the concepts of autonomous computing, such self-configuration and self-adaptation, may enhance system performance instantly. The advancement of autonomic database management will depend critically on addressing issues with mixed workload types and expanding the methodology to accommodate greater datasets and various database systems.

6 REFERENCE

[1] Yu, X., Yu, X., & Zhang, X. (2021). Case-based reasoning adaptation based on fuzzy gravitational search algorithm for disaster emergency plan. Journal of Intelligent & Fuzzy Systems, 40(6), 11007-11022.

[2] Feng, L., Hou, Y., & Zhu, Z. (2021). Optinformatics in Evolutionary Learning and Optimization (Vol. 25). Springer Nature.

[3] Delisle, J. F. (2022). Intelligent Adaptive Flight Training System-A Human Performance in the Loop for Real-Time Decision Making (Doctoral dissertation, Polytechnique Montréal).

[4] Zhao, Z., Chen, J., Yao, J., Xu, K., Liao, Y., Xie, H., & Gan, X. (2023). An improved spatial case-based reasoning considering multiple spatial drivers of geographic events and its application in landslide susceptibility mapping. Catena, 223, 106940.

[5] Yang, Z. (2022). Optimal loading method of multi type railway flatcars based on improved genetic algorithm. Journal of Intelligent Systems, 31(1), 915-926.

[6] Biswas, P. K., & Liu, S. (2022). A hybrid recommender system for recommending smartphones to prospective customers. Expert Systems with Applications, 208, 118058.

[7] Louis, L. E. L., Moussaoui, S., Van Langhenhove, A., Ravoux, S., Le Jan, T., Roualdes, V., & Milleville-Pennel, I. (2023). Cognitive tasks and combined statistical methods to evaluate, model, and predict mental workload. Frontiers in Psychology, 14, 1122793.

[8] Shaheen, N., Raza, B., Shahid, A. R., & Malik, A. K. (2021). Autonomic workload performance modeling for large-scale databases and data warehouses through deep belief network with data augmentation using conditional generative adversarial networks. IEEE Access, 9, 97603-97620.

[9] Genkin, M. (2021). Autonomic Performance Optimization of Big Data Workloads (Doctoral dissertation, Carleton University).

[10] Feng, B., Ding, Z., & Jiang, C. (2022). FAST: A forecasting model with adaptive sliding window and time locality integration for dynamic cloud workloads. IEEE Transactions on Services Computing, 16(2), 1184-1197.