Journal of AI and Data Mining
Vol 8, No 4, 2020, 471-480. ***Original Research Paper*** DOI: 10.22044/JADM.2020.9403.2076

A Monte Carlo-Based Search Strategy for Dimensionality Reduction in Performance Tuning Parameters

A. O. Omondi[1*], I. A. Lukando[1] and G. W. Wanyembi[2]

1. Faculty of Information Technology, Strathmore University, Nairobi, Kenya.
2. Department of Information Technology, Mount Kenya University, Thika, Kenya.

Received 20 February 2020; Accepted 30 March 2020

**Corresponding author: aomondi@strathmore.edu (A. O. Omondi).*

Abstract

Redundant and irrelevant features in dimensional data increase the complexity in the underlying mathematical models. It is necessary to conduct pre-processing steps that search for the most relevant features in order to reduce the dimensionality of the data. This work makes use of a meta-heuristic search approach that uses lightweight random simulations to balance between the exploitation of relevant features and the exploration of features that have the potential to be relevant. In doing so, this work evaluates how effective the manipulation of the search component in feature selection is on achieving a high accuracy with reduced dimensions. A control group experimental design is used in order to observe the factual evidence. The context of the experiment is the high-dimensional data experienced in the performance tuning of complex database systems. The Wilcoxon signed-rank test at the .05 level of significance is used to compare the repeated classification accuracy measurements on the independent experiment and control group samples. Encouraging results with a p-value < 0.05, were recorded and provided evidence to reject the null hypothesis in favour of the alternative hypothesis, which states that the meta-heuristic search approaches are effective in achieving a high accuracy with reduced dimensions depending on the outcome variable under investigation.

Keywords: *Dimensionality Reduction, Meta-heuristic Search, Monte Carlo, Performance Tuning, Reinforcement Learning, Database Systems, System Administration.*

1. Introduction

Data analysis is a common activity in today's data-driven world. Most datasets are high-dimensional datasets with hundreds or thousands of features. However, a substantial number of these features are either redundant or irrelevant [1]. It is necessary to select the most significant features as a mandatory preliminary step before any predictive or classification task is performed. It is also important to ensure that this selection of features does not damage the underlying structure of the dataset. Dimensionality reduction can be used to achieve this.

It is a common practice to divide dimensionality reduction into 2 steps: feature extraction and feature selection. Feature extraction involves transforming or projecting a space composed of many dimensions into a space composed of fewer dimensions, whereas feature selection involves the process of selecting only the relevant and non-redundant features. The feature selection step is subsequently made up of 2 components: a search component and an evaluator component. It is the search component that is responsible for generating the candidate feature subsets whereas the evaluator component checks the goodness of the candidate feature subsets. Examples of feature selection methods include the filter, wrapper, and hybrid methods. The recent literature also includes the embedded, ensemble, and integrative methods [2]. Examples of the approaches that can be applied in the search component include the exhaustive, heuristic, and meta-heuristic search approaches.

Statistical tests for correlation with the outcome variable provide a score that is used to select features in filter methods. The selection is, therefore, independent from any machine learning algorithm. Filter methods are classifier independent and use some proxy measures to evaluate the features. Examples of statistical tests used in filter methods include the Pearson's correlation coefficient, linear discriminant analysis, analysis of variance, and Chi-square tests.

Wrapper methods use a subset of features and train a model using them. Unlike filter methods, which use statistical methods to evaluate a subset of features, wrapper methods use cross-validation. Wrapper methods are classifier-dependent and use the classifier directly to score the feature subsets. Forward feature selection, backward feature elimination, and recursive feature elimination are examples of the techniques that can be used in wrapper methods.

Embedded methods are implemented by the algorithms that have their own in-built feature selection methods. They work by adding a penalty against complexity to reduce the degree of overfitting or variance of a model. It does not separate the learning and feature selection modules. Embedded methods take advantage of both the filter and wrapper methods using an independent metric to rank the features and using a learning algorithm to quantify the strength of feature subsets. Examples include LASSO regression, ridge regression, regularized trees, memetic algorithms, and random multinomial logit. In addition to the above 3 standard methods, we also have hybrid methods, which incorporate multiple types of feature selection methods within the same process. Ensemble methods combine independent feature subsets and could eventually provide a better approximation to the optimal subset of features. Integrative methods integrate the external data during the process of feature selection.

This work explores meta-heuristic search approaches in the feature selection step of dimensionality reduction. This is done independent from the classifier algorithms as is the case with filter methods, which select features based on their correlation with the outcome variable. Meta-Heuristic search approaches use lightweight random simulations to find the best solutions. An example of an algorithm that applies a meta-heuristic search is the Monte Carlo Tree Search algorithm as shown in Figure 1.

They also combine the exploitation of good solutions with the exploration of new ones. This enables them to continuously tend towards reaching a globally optimum solution.

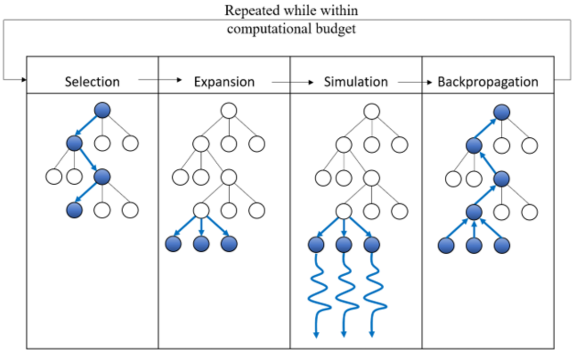

Figure 1. Monte Carlo Tree Search.

An in-depth explanation of this search technique has been provided in our previous work [3]. Figure 2 depicts the research paradigm that puts together the conceptual framework of the current work.

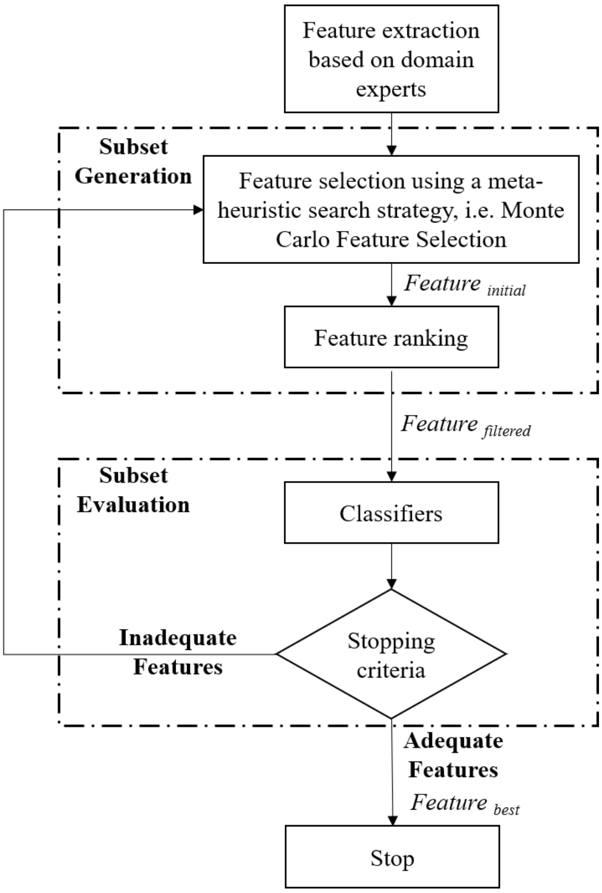

Figure 2. Research paradigm.

The objective of this research work was to evaluate how effective the use of a meta-heuristic search approach in feature selection is on achieving a high accuracy with reduced dimensions. The following questions provided a point of departure to carry out the research work:

(i) Guiding question: What type of variations can be implemented in a dimensionality reduction algorithm?

(ii) Experimental question: How can a Monte Carlo based search strategy be applied in a dimensionality reduction algorithm?

(iii) Categorization question: How can the effectiveness of a dimensionality reduction technique be measured?

Section 2 of the paper specifies the methodology that describes how the research work was conducted. This is followed by Section 3, which presents the results observed when conducting the experiment. Section 4 discusses what the results imply, and Section 5 concludes the paper.

2. Methodology

Differences in philosophical perspectives combined with the questions that a research work seeks to answer, to a large extent, determine the method of enquiry. The work, being scientific in nature, makes the philosophical assumption that material things are more real than immaterial phenomena. According to the principles of ontological materialism, this assumption implies that reality exists regardless of the researcher's intuitive perception. With such an assumption in mind, the method used to justify a claim as true is critical to both the validity and reliability of the research work [4]. The research work subsequently places an emphasis on using observations in form of input from sensors to justify a claim as true. This approach is classified as having a positivistic epistemology. An inductive-deductive scientific approach is necessary for a positivistic epistemology to be applied. The philosophical perspective and epistemology of this research work, therefore, justifies the use of experiments to obtain the observations required to reveal true answers to the research work questions. It is important to note that the research work questions focus on an inanimate object that has a reality that is distinct from the social actors that interact with it. This warrants the need for an objective, rather than a subjective, approach. The specific type of experiment design that has been applied is a control group experimental design, as explained further in Section 2.1.

2.1. Experiment Procedure

A cross-sectional study was appropriate in this context, considering that the study was not focused on measuring the change over time. Instead, the study was keen on introducing an intervention (in the form of the independent variable) and retrospectively confirming if it has produced the desired effect (on the dependent variable).

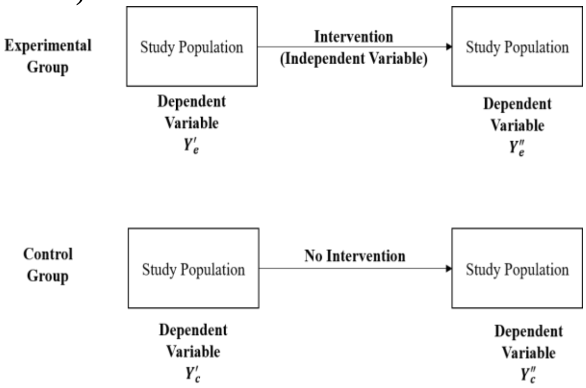
Experimental Group
Study Population
Intervention (Independent Variable)
Study Population
Dependent Variable Y'_e
Dependent Variable Y''_e
Control Group
Study Population
No Intervention
Study Population
Dependent Variable Y'_c
Dependent Variable Y''_c

Figure 3. Control group experimental design.

Application of a control group experimental design supports this approach. Figure 3 provides a graphical representation.

One of the key advantages of a control group experimental design is its ability to quantify the impact of extraneous variables. This is an important requirement needed to guarantee the internal validity of the research work by ensuring that the effect on the dependent variable is caused solely by the independent variable and not by the extraneous variables.

Negative effects common to this experimental design such as the maturation effect (changes observed since the population is maturing over time), the reactive effect (where the pre-test data affects the post-test data since the instrument educates the respondents), and the regression effect (where the respondents shift their stance from an extreme end to a mid-point over time) were not experienced. This is because the focus of the research work was not on the dynamic and subjective human actors, for example, database administrators, which interact with the inanimate object. Experiment E_1 stipulates the procedure followed to implement an inductive-deductive scientific approach.

Experiment E_1

Goal: To measure the effect of applying a Monte Carlo-based search strategy in the search component of a dimensionality reduction algorithm.

Null Hypothesis: Applying a Monte Carlo-based search strategy to search for the relevant and non-redundant features in the search component of a dimensionality reduction algorithm has no positive impact on the accuracy obtained when using the selected features to perform a classification task

Independent variable: Dimensionality reduction algorithm.

Dependent variable: Classification accuracy

Data Analysis to be performed: The Wilcoxon signed-rank test at the .05 level of significance to compare the repeated classification accuracy measurements on the independent experiment and control group samples.

Procedure

Table 1. Experiment Procedure.

Step	Description
I	Load the sample dataset into the test bed
II	**If** the test bed is setup for the TPS experimental group, apply a meta-heuristic search strategy (a Monte Carlo-based search strategy) to the experimental group to identify the most relevant and least redundant features that can be used to classify the performance of the Database System (DBS) as measured by “max_transaction_throughput_TPS_y_1” (y1). **If** the test bed is setup for the response time experimental group, apply a meta-heuristic search strategy (a Monte Carlo-based search strategy) on the experimental group to identify the most relevant and least redundant features that can be used to classify the performance of DBS as measured by “avg_response_time_seconds_y_2” (y2). **Else, if** the test bed is setup for the control group, randomly select features from the feature set partly informed by intuition and DBA best-practices. The result of this step should be an initial set of features ($F_{original}$).
III	Apply an entropy-based feature ranking algorithm (Symmetrical Uncertainty algorithm) to select the top 3 features from $F_{original}$ based on their relative importance. The result should be a filtered set of features ($F_{filtered}$).
IV	Apply classification algorithms that use the filtered set of features ($F_{filtered}$) to classify observations in the dataset. The result should be a classification on whether the database system’s performance is good or bad as measured by either y1, y2, or “avg_transaction_time_seconds_y_3” (y3).
V	Record the classification accuracy and the Cohen’s Kappa statistic for each one of the classification algorithms
VI	Apply the Wilcoxon signed-rank test at the .05 level of significance to compare the classification accuracy and Cohen’s Kappa statistics on the independent experiment and control group samples.

2.2. Experiment Dataset

Performance tuning data was generated using the TPC-E benchmark. TPC-E contains a mix of 10 concurrent transactions of different types and complexity executed either online or queried for deferred execution. The 10 concurrent transactions were broker-volume, customer-position, market-feed, market-watch, security-detail, trade-lookup, trade-order, trade-result, trade-status, and trade-update transaction. Each one of the 10 transactions was assigned the same weight without any latency between their executions. This means that the transactions queried for deferred execution had the same priority. The database itself was comprised of 33 tables with a wide range of record and population sizes totalling to 12.79 GB of data.

Each one of the 10 transactions was executed an average of 40 times in each run. There was a total of 10 runs with each run having a set of different parameter configurations informed by the domain experts in the literature reviewed. This yields between 3,900 and 4,100 transactions executed with different parameter configurations. There were 3 outcome variables that had the potential to be used as indicators to classify the overall performance of the database system based on various parameter configurations. The 3 outcome variables were “max_transaction_throughput_TPS_y_1” (y1), “avg_response_time_seconds_y_2” (y2), and “avg_transaction_time_seconds_y_3” (y3). The outcome of the classifier was either “good” performance or “bad” performance defined by either the Transactions Per Second (TPS), response time (seconds), or the transaction time (seconds).

2.3. Experiment Test Bed

The study made use of a pre-defined test bed to conduct experiments. R (version 3.5.0), an integrated suite of software facilities for statistical computation, data manipulation, and graphical display were used to conduct the experiment [5]. RStudio (version 1.1.463), an Integrated Development Environment (IDE), was used to provide comprehensive facilities that make using R easier. The hardware platform was made up of a 64-bit Windows 10 Operating System, with 16GB of RAM, and a 2.6GHz Intel® Core™ i7 processor. The test bed setup for the experimental group applied a meta-heuristic search strategy (a Monte Carlo based search approach) before the entropy-based feature ranking stage, whereas the control group setup did not apply any meta-heuristic search strategy. The study made use of four existing R packages to conduct the experiment: ‘FSelectorRcpp’, ‘e1071’, ‘caret’, and ‘readr’. Firstly, the ‘FSelectorRcpp’ R package (version 0.3.1) was used to obtain the functions required to minimize the redundant and irrelevant features in a given dataset [6]. It provided a Java/Weka-free implementation that contained entropy-based filters such as Gain Ratio and Symmetrical Uncertainty.

Table 2. List of features.

Feature	Identified as relevant and non-redundant through the meta-heuristic search approach
innodb_buffer_pool_size__M_x_1	Yes
innodb_buffer_pool_instances_x_2	Yes
innodb_old_blocks_pct_x_3	Yes
innodb_old_blocks_time__ms_x_4	No
innodb_buffer_pool_dump_at_shutdown__IO_x_5	No
innodb_buffer_pool_load_at_startup__IO_x_6	No
query_cache_size__M_x_7	No
innodb_log_file_size__M_x_8	Yes
innodb_file_per_table__IO_x_9	No
innodb_lock_wait_timeout__SEC_x_10	No
thread_cache_size_x_11	No
tmp_table_size__M_x_12	Yes
max_heap_table_size__M_x_13	Yes
sort_buffer_size__K_x_14	No
join_buffer_size__K_x_15	No

The study applied Symmetrical Uncertainty as a baseline to perform an entropy-based feature ranking on the selected features for all tests. Secondly, the ‘e1071’ R Package (version 1.7-2) was used to obtain the statistical computation functions required for classification [7]. The study applied the 10 classifiers for all tests as listed in Section 3.2.

Thirdly, the ‘caret’ (Classification and Regression Training) R package (version 6.0-84) was used to obtain the functions required for plotting the classification and regression models [8]. The confusion matrix function in the ‘caret’ package was used extensively to make observations during the experiment.

Lastly, the ‘readr’ (Read Rectangular Text Data) R package (version 1.3.1) was used to obtain the functions required to read rectangular data from flat files in different formats such as the comma separated value ‘csv’ or the tab separated value ‘tsv’ formats [9]. The study made use of the ‘read_csv()’ function in the ‘readr’ package to read and load the data set.

2.4. Data Analysis Method

The study applied the non-parametric Wilcoxon signed-rank test at the .05 level of significance to compare repeated accuracy measurements on the independent experiment and control group samples. This was done by assessing how significantly their means differ. The Wilcoxon signed-rank test can be used as an alternative to the paired student’s t-test when the sample size is small and there is no guarantee that the population is normally distributed, which was the case in this study.

3. Results

Following the experiment procedure outlined in Section 2.1, the meta-heuristic search strategy based on a Monte Carlo Tree Search identified 6 relevant and non-redundant features out of the initial 15 features. It is important to note that there are hundreds of features representing parameters that can be tuned on a MySQL (version 8.0) standalone database system.

The list of 15 features was identified through common performance tuning best practices in the IT industry. Table 2 presents the list of features that could be manipulated in order to observe their effect on the 3 outcome variables (y1, y2, and y3). Out of the 6 relevant and non-redundant features, the research work went a step further to select those that had the highest rank. This was done while maintaining the objective of achieving a high accuracy with reduced dimensions. Section 3.1 presents the feature ranking results.

Table 3. Experiment results

	Experimental Group: TPS, %		**Experimental Group: Response Time, %**		**Control Group: Random Selection, %**	
Classifier	**Accuracy**	**Kappa**	**Accuracy**	**Kappa**	**Accuracy**	**Kappa**
Logistic Regression (glm)	0.99	0.99	0.65	0.28	0.67	0.35
Lasso and Elastic-Net Regularized Generalized Linear Model (glmnet)	0.99	0.99	0.64	0.27	0.67	0.35
Support Vector Machines with Radial Basis Function Kernel (svmRadial)	0.99	0.99	0.79	0.58	0.67	0.35
k-Nearest Neighbors (knn)	0.99	0.99	0.79	0.58	0.67	0.35
Naïve Bayes (nb)	0.88	0.76	0.76	0.53	0.67	0.35
Recursive partitioning for classification, regression and survival trees (rpart)	0.99	0.99	0.80	0.60	0.67	0.35
C5.0 Decision Trees and Rule-Based Models (c5.0)	0.99	0.99	0.80	0.60	0.67	0.35
Bootstrap Aggregation (Bagging) for classification, regression and survival trees (CART) (treebag)	0.99	0.99	0.78	0.55	0.67	0.35
Random Forest (rf)	0.99	0.99	0.78	0.55	0.67	0.35
Stochastic Gradient Boosting (Generalized Boosted Modelling) (gbm)	0.99	0.99	0.79	0.58	0.67	0.35

3.1. Relative Importance of Features

Figure 4 shows the relative importance of the independent features according to Symmetrical Uncertainty. Figure 5 presents the relative importance of the independent features according to the Monte Carlo Feature Selection.

3.2. Accuracy and Cohen's Kappa Metrics

The 3 dependent variables that directly indicated the performance level were "max_transaction_throughput_TPS_y_1" (y1), "avg_response_time_seconds_y_2" (y2), and "avg_transaction_time_seconds_y_3" (y3).

A -0.20 correlation between y1 and y2 and a -0.24 correlation between y1 and y3 were noted, whereas a 0.99 correlation between y2 and y3 was noted.

The research work, therefore, had 2 experimental groups. The first experimental group used y1 as the dependent variable. Since the correlation between y2 and y3 was high, either of the 2 could be used. The research work used y2 as the dependent variable in the second experimental group.

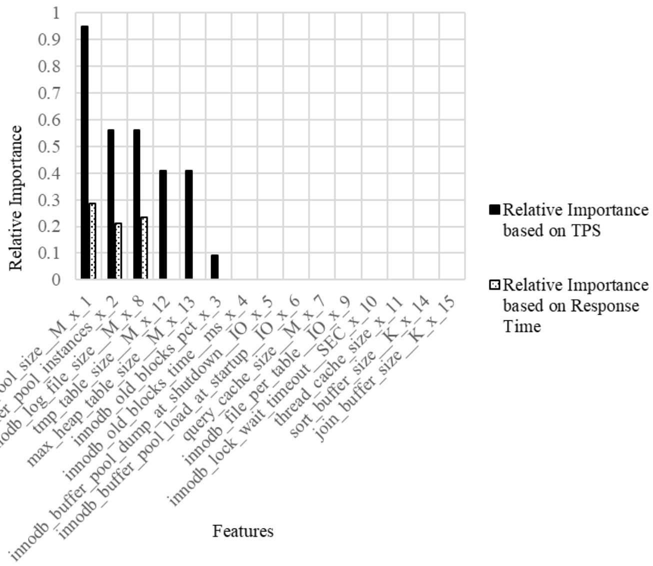

Figure 4. Relative importance of features according to information gain (Symmetrical Uncertainty).

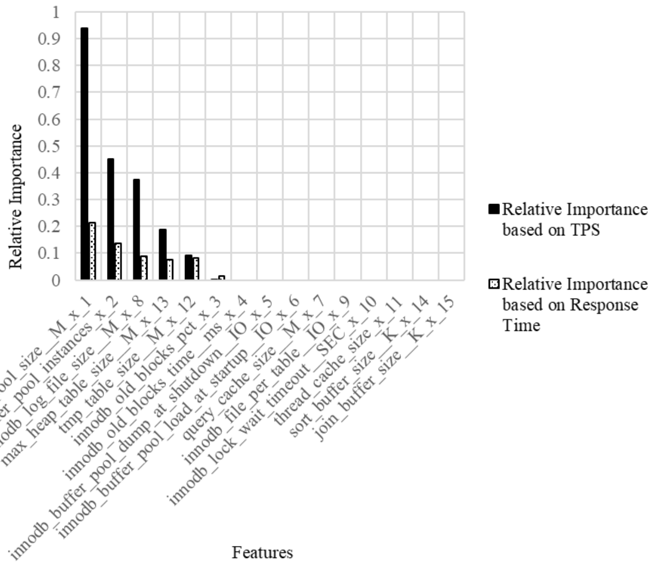

Figure 5. Relative importance of features according to Monte Carlo Feature Selection (MCFS).

Figure 6, Figure 7, and Figure 8 present a graphical representation of the results, showing dot plots that display the spread of accuracy and the Cohen's kappa statistic. Figure 7, which applied the meta-heuristic search algorithm, shows the highest accuracy results compared to the random selection of features presented in Figure 6. Figure 8, on the other hand, provides evidence that it is more efficient to use Transactions per Second (TPS), as opposed to the response time latency, to classify the performance of a database system.

3.3. Wilcoxon Signed Rank Test Results

The p-value of the TPS experimental group versus the control group was 2.428e-05, whereas the p-value of the response time experimental group versus the random selection control group was 0.0171. We, therefore, reject the null hypothesis in both experiments in favour of the alternative hypothesis.

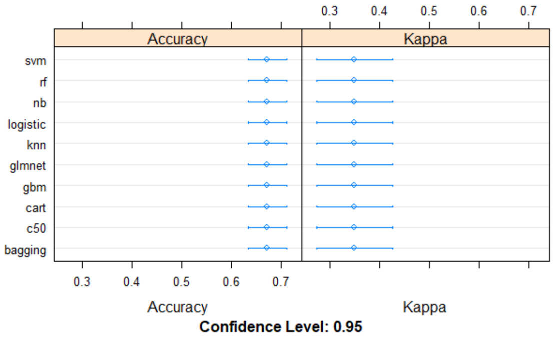

Figure 6. Experiment results with randomly selected features.

The alternative hypothesis states that applying a Monte Carlo-based search strategy to search for the relevant and non-redundant features in the search component of a dimensionality reduction algorithm has a positive impact on the accuracy obtained when using the selected features to perform a classification task.

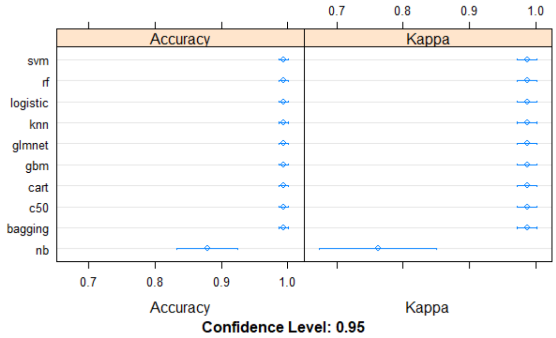

Figure 7. Experiment results for the TPS experimental group.

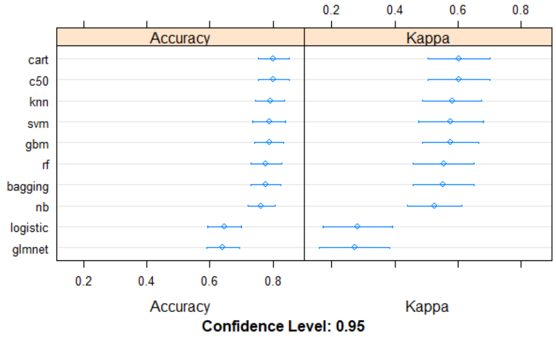

Figure 8. Experiment results for the response time experimental group.

4. Discussion

A profile of a system provides the useful information required to understand its current state. It contains the multiple features and numerous values for each feature, thus increasing the dimensionality of a data set that constitutes it. However, there are features that are redundant and irrelevant depending on the analysis that is required to be performed. Selecting the redundant and irrelevant features is computationally expensive and complicates the whole analysis process [2]. This poses a significant challenge to the system administrators who conduct performance tuning in complex systems.

The importance of search algorithms to identify the relevant and non-redundant features is highlighted at this point. The study hypothesized that applying a Monte Carlo-based search strategy (meta-heuristic search) to search for the relevant and non-redundant features in the search component of a dimensionality reduction algorithm had a positive impact on the accuracy obtained when using the selected features to perform a classification task. The hypothesis was subjected to scrutiny through a control group experimental design. The experimental results were analyzed to establish their statistical significance, and subsequently, formed a basis to either accept or reject the hypothesis.

The results obtained provide evidence that it is better to use a Monte Carlo-based search strategy to reduce the dimensionality of a data set. These results can be used in a wide range of possible application areas that involve data analysis. In this research work, the benefits of the results were highlighted in performance tuning to identify the parameters that, if tuned by a system administrator, will have the greatest impact on the system's overall performance.

It was expected that applying a meta-heuristic search approach in the feature selection technique and entropy-based information gain (symmetrical uncertainty) in the feature ranking technique would lead to a higher classification accuracy. The results of the current study confirmed that this was indeed the case; however, it was beyond the scope of this study to examine the algorithms that implement other search approaches, for example, the exhaustive and heuristic search approaches. The experiment was, therefore, setup to compare the use of a meta-heuristic search approach versus not using a meta-heuristic search approach, i.e., selecting the features to tune based on the DBA intuition and best practices.

A positive correlation between y2 and y3 was interesting, but not surprising. While y2 measured the query response time in s, y3 measured the transaction response time in s. The distinction between a read-intensive query and a write-intensive transaction is not clearly defined, particularly in complex queries. Complex queries would usually involve a hybrid of reading and writing data, as was the case with the TPC-E transactions used in the experiment. It is on this basis that the research chose to use either of the two variables, as opposed to all the two, as the outcome variables in the experiment. Specifically, y2 was used in the second experimental group. Similar results were observed when y2 was the outcome variable; however, the classification accuracy was lower than when y1 was the outcome variable.

It was surprising to note that there was no correlation between y1 and y2 and between y1 and y3. The evidence from this work suggests that measuring the performance based on Transactions per Second (y1) is ideal for specific contexts that involve an Online Transaction Processing (OLTP), system, whereas measuring the performance based on either query response time (y2) or transaction response time (y3) is ideal for Online Analytical Processing (OLAP) systems. We have explained this observation further in our recent paper [10]. In general, the current study found that using Transactions per Second (y1), as an outcome variable, yields a higher classification accuracy, as opposed to using query response time (y2) as the outcome variable. Overall, the study strengthens the idea that measuring the performance of a database system based on only Transactions per Second leads to a higher accuracy of classifiers.

The Monte Carlo-based meta-heuristic search approach for feature selection and the entropy-based feature ranking identified 6 features as the most relevant and least redundant. The 6 features were innodb_buffer_pool_size__M_x_1, innodb_buffer_pool_instances_x_2, innodb_old_blocks_pct_x_3, innodb_log_file_size__M_x_8, tmp_table_size__M_x_12, and max_heap_table_size__M_x_13. This means that manipulating these 6 features within their pre-defined constraints will have the highest impact on the performance of the database system as measured by “max_transaction_throughput_TPS_y_1” (y1). The database administrators should therefore not waste their effort in tuning the remaining hundreds of features that are either redundant and/or irrelevant.

High-dimensional data require intense processing due to the complexity, heterogeneity, and hybridity of the features involved. The feature selection methods can help to substantially reduce the complexity of data, thus making it easier to analyse and transform into useful information [11]. The current work provides evidence that indicates that the search component is an important module in a dimensionality reduction algorithm. A meta-heuristic search that exploits known good options while exploring possible good options is ideal.

5. Conclusion

This work set out to evaluate how effective the use of a meta-heuristic search approach in feature selection is on achieving a high accuracy with reduced dimensions. The experiment confirmed that using a Monte Carlo-based feature selection technique followed by an entropy-based feature ranking technique led to the selection of features that were relevant and not redundant, and ultimately to a higher classification accuracy. Overall, the work strengthens the idea that performing feature selection as a pre-processing step makes the underlying models simpler and easier to understand. It also makes the data analysis computationally affordable. Although this work focuses on performance tuning, the findings may well have a bearing on any data analysis that involves high-dimensional data. A good example is genomic data, which is becoming more common in the bioinformatics field [2].

This work lays the groundwork for future research works into search approaches that can be used to identify the most relevant and least redundant features in high-dimensional datasets. Other than the Monte Carlo feature selection, other search approaches that can be used include genetic algorithms, particle swarm optimization, bat algorithms, ant colony optimization, and multi-objective evolutionary algorithms. In terms of directions for future research, further work could investigate on algorithms that apply exhaustive search and heuristic search approaches.

Although automation has supported the redesign of many business processes, it is fragile and cumbersome. The findings presented in this work shed new light on the use of autonomous systems. These are systems that can manage themselves given high-level objectives from the administrators [12]. In this scenario, a system can search for the most relevant and least redundant features to tune by itself and through reinforcement learning, discover the most rewarding ways to tune these features. This is a concept that we investigated further in the context of the Othello computer game in our previous work [3]. A reasonable approach to promote autonomic computing is to have horizontal building blocks and not vertical building blocks. This enables the different modules to be used in diverse application areas.

References

[1] Chaudhry, M. U. & Lee, J. (2018). Feature Selection for High Dimensional Data Using Monte Carlo Tree Search, IEEE Access, vol. 6, pp. 76036-76048, 2018, doi: 10.1109/ACCESS.2018.2883537.

[2] Tadist, K., Najah, S., Nikolov, N. S., Mrabti, F. & Zahi, A. (2019). Feature Selection Methods and Genomic Big Data: A Systematic Review, Journal of Big Data, vol. 6, no. 1, p. 79, Aug. 2019, doi: 10.1186/s40537-019-0241-0.

[3] Omondi, A. O., Lukandu, I. A. & Wanyembi, G. W. (2019). A Variated Monte Carlo Tree Search Algorithm for Automatic Performance Tuning to Achieve Load Scalability in InnoDB Storage Engines’, IRJAES, vol. 4, no. 1, pp. 100-110.

[4] Hjørland, B. (2005). Empiricism, rationalism and positivism in library and information science', Journal of documentation, vol. 61, no. 1, pp. 130-155, 2005, doi: 10.1108/00220410510578050.

[5] R Core Team, R: A Language and Environment for Statistical Computing. Vienna, Austria: R Foundation for Statistical Computing, 2018.

[6] Zawadzki, Z. & Kosinski, M. (2019). FSelectorRcpp: 'Rcpp' Implementation of 'FSelector' Entropy-Based Feature Selection Algorithms with a Sparse Matrix Support. 2019.

[7] Meyer, D., Dimitriadou, E., Hornik, K., Weingessel, A. & Leisch, F. (2019). e1071: Misc Functions of the Department of Statistics, Probability Theory Group (Formerly: E1071), TU Wien. 2019.

[8] Wing, M. K. C. from J. et al., caret: Classification and Regression Training. 2019.

[9] Wickham, H., Hester, J. & Francois, R. (2018). readr: Read Rectangular Text Data. 2018.

[10] Omondi, A. O., Lukandu, I. A. & Wanyembi, G. W. (2018). Scalability and Nonlinear Performance Tuning in Storage Servers', IJRSSET, vol. 5, no. 9, pp. 7-18, Nov. 2018.

[11] Zhao, Z., Morstatter, F., Sharma, S., Alelyani, S., Anand, A. & Liu, H. (2010). Advancing feature selection research, ASU Feature Selection Repository Arizona State University, pp. 1-28, 2010.

[12] Kephart, J. O. & Chess, D. M. (2003). The vision of autonomic computing', Computer, vol. 36, no. 1, pp. 41-50, 2003.

IAIDM

نشریه هوش مصنوعی و داده کاوی

دانشگاه صنعتی شاهرود

یک استراتژی جستجوی مبتنی بر مونت کارلو برای کاهش ابعاد در پارامترهای تنظیم عملکرد

آلان اودیامبو اومندی[۱،*] ، اسماعیل آیتا لوکاندو[۲] و گریگوری وابوکه وانیمبی[۳]

[۱] دانشکده فناوری اطلاعات، دانشگاه استراتور، نایروبی، کنیا.

[۲] گروه فناوری اطلاعات، دانشگاه مونت کنیا، تیکا، کنیا.

ارسال ۲۰۲۰/۰۲/۲۰؛ پذیرش ۲۰۲۰/۰۳/۳۰

چکیده:

ویژگی‌های زائد و بی‌ربط در داده‌های بعدی، پیچیدگی در مدل‌های اساسی ریاضی را افزایش می‌دهد. انجام مراحل پیش پردازش برای جستجوی مناسب‌ترین ویژگی‌ها به منظور کاهش ابعاد داده‌ها ضروری است. این کار از رویکرد جستجوی فرا-ابتکاری استفاده می‌کند که از شبیه سازی‌های تصادفی سبک برای تعادل بین بهره‌برداری از ویژگی‌های مربوطه و اکتشاف ویژگی‌های بالقوه مرتبط استفاده می‌کند. این کار ارزیابی می‌کند که دستکاری مولفه جستجو در انتخاب ویژگی در دستیابی به دقت بالا با ابعاد کاهش یافته چقدر موثر است. به منظور مشاهده شواهد واقعی، از یک طرح آزمایشی گروه کنترل استفاده شده است. زمینه این آزمایش داده‌های با ابعاد بالا است که در تنظیم عملکرد سیستم‌های پایگاه داده پیچیده تجربه شده است. آزمون نمره امتحان شده Wilcoxon در سطح معناداری ۰/۰۵ برای مقایسه اندازه گیری‌های مکرر دقت طبقه‌بندی در آزمایش‌های مستقل و نمونه‌های گروه کنترل مورد استفاده قرار می‌گیرد. نتایج تشویقی با p-value <0.05 ثبت و شواهدی ارائه شد که فرضیه صفر را به نفع فرضیه جایگزین رد می‌کند، که بیان می‌کند روشهای جستجوی فرا-ابتکاری در دستیابی به دقت بالا با کاهش ابعاد بسته به نتیجه متغیر تحت بررسی موثر است.

کلمات کلیدی: کاهش ابعاد، جستجوی فرا-ابتکاری، مونت کارلو، تنظیم عملکرد، یادگیری تقویتی، سیستم‌های پایگاه داده، مدیریت.