ISSN: 2616 - 6127

AZERBAIJAN JOURNAL OF HIGH PERFORMANCE COMPUTING

Volume 6, Issue 2

December 2023

Sponsored by ADNSU

Azerbaijan Journal of High Performance Computing, Vol 6, Issue 2, 2023, pp. 135-152
https://doi.org/10.32010/26166127.2023.6.2.135.152

Predicting the Status of Thyroid and Cardiovascular Patients According to Their Electronic Records Using Temporal Elements Based on the Combination of Shuffled Frog Leaping Algorithm (SFLA) and Deep Learning

Amirhossein Jalilzadeh Afshari

Azad University of Zanjan, Zanjan, Iran, amir.jalilzadeafshar@gmail.com

*Correspondence: Amirhossein Jalilzadeh Afshari, Azad University of Zanjan, Zanjan, Iran, amir.jalilzadeafshar@gmail.com

Abstract

Health and treatment are two of the most important application fields of information technology, in which the problem of predicting a disease is highly important. The physician makes such predictions based on the clinical condition of the patient and the level of facilities and advances in medical knowledge for the patient—information technology benefits from multiple methods to help this field. Accordingly, the patient information storage system, drug information, treatment and surgery systems, treatment follow-up systems, remote treatment systems, etc., aim to facilitate the treatment process. The patient can receive the best services within the shortest time due to these systems and information availability. The doctor can provide services to his patient anywhere in the world. This paper provided a model to predict the condition of patients based on their electronic records using temporal elements based on combining the shuffled frog leaping algorithm (SFLA) and deep learning. Accordingly, the evolutionary shuffled frog leaping algorithm (SFLA) and deep learning were used for preprocessing, feature selection, and classification. Two datasets of cardiovascular and thyroid diseases were utilized in the simulation section to ensure the efficiency of the proposed method. Based on this simulation, the proposed method indicated improvement compared to similar methods in the evaluated datasets. In the cardiovascular diseases dataset, this improvement was recorded as 1.4% and 3.2% compared to the author's previous and updated similar methods, respectively.

Keyword: Prediction of Patients' Conditions, Electronic File, Shuffled Frog Leaping Algorithm (SFLA), Deep Learning.

1. Introduction

According to WHO, health care quality is defined as a level of health services provided for individuals and populations that enhances the likelihood of optimal health outcomes.

Due to the advancement of digital technology, most tasks and work in the healthcare sector are being digitized and well-organized, which may dramatically improve the quality of healthcare services compared to the traditional approach. The Electronic Health Record (EHR) system appears at the forefront of implementation in healthcare institutions to enhance healthcare measure quality. EHR systems enable data-based clinical decision-making to improve the quality of healthcare. According to Gattiti et al., the accurate adoption of EHR systems can improve healthcare quality by increasing patients' safety and ensuring effective, efficient, timely, fair, and patient-centered care. Despite the advantages of EHR systems, problems or unintended outcomes prevent the acceptance and successful utilization of EHR systems in healthcare settings. Some of the most common of these factors are as follows: Physician burnout, failure of expectations, EHR market saturation, absences of innovation, data obfuscation, interoperability, privacy in sharing data, process prolonging until the completion of tasks, interruption in accomplishing tasks and solutions at the point of care, and non-coordination between technology and clinical context (Woldemariam, M. T., & Jimma, W., 2023). Disease prediction can benefit stakeholders such as the government and health insurance companies. It is capable of identifying patients at risk of diseases or health conditions. Subsequently, doctors can take appropriate measures to prevent or minimize the risk and, in turn, improve the quality of care and avoid potential hospital admissions. Also, considering the recent advancements in data analysis tools and techniques, disease risk prediction would be able to use large amounts of semantic information such as demographics, clinical diagnoses and measurements, health behaviors, laboratory results, prescriptions, and use of care measures (Hossain et al.; S., 2019).

Electronic health data are computerized medical records of patients that contain information about healthcare institutions. These data refer to diseases or conditions of the patient and are recorded in electronic systems with the initial goal of providing relevant health care and services. Administrative Healthcare Data, Administrative Claims Data, Computerized Claims Data, Digital Health Records, or Electronic Health Records are all used to describe electronic health data. Electronic health data are rapidly used for modeling and decision-making in the health care research section. This data type is used beyond keeping records in the health care research section. For example, they are used to analyze healthcare utilization, monitor the hospital care network's effectiveness, and develop predictive models for disease prediction (Lu et al., S., 2023, April).

Data mining is a systematic method to extract valuable and meaningful patterns from big data. It is a process that discovers unknown patterns and trends in data stores. Such information is mainly used to make prediction models. Prediction models play a substantial role in the healthcare sector. Different approaches and models help reduce human efforts to observe and quantify the relationships among the various features, patterns, colored graphs of healthcare datasets, etc. (Mulla, F. D., & Jayakumar, N., 2018, November). Machine-learning and deep-learning approaches have recently been used in data-driven healthcare research. Many supervised machine-learning algorithms have been utilized for

risk assessment by disease risk prediction models.

Similarly, using deep learning methods has brought remarkable advances in health informatics. Such models can efficiently capture and record complicated relationships between high-dimensional features through hierarchical levels of manipulation when used to train a prediction model. For instance, the convolutional neural network performs exceptionally well in visual medical image analysis. In addition, recurrent neural networks provide exceptional accuracy in language processing through the recurrent neural network architecture (Lu, H., & Uddin, S., 2023, April).

The accuracy and reliability of risk assessment models mainly depend on predictors and methods of development, validation, calibration, and clinical application. Administrative data are limited due to not having clinical specificity in choosing a proper set of predictors. The utilization of machine learning methods in medicine has also developed for laboratory conditions and results with recent multi-billion dollar investments in electronic medical records and their ever-increasing use and application in healthcare systems. Thus, an increase has emerged in the development of highly sophisticated prediction models using EMR over the last few years (Mahmoudi, E., Kamdar N. et al., 2020). This paper focused on predicting the health status of patients according to their electronic records using temporal elements based on the evolutionary shuffled frog leaping algorithm (SFLA) and deep learning technique.

2. Relevant literature

Mahmoudi et al. provided a model using the data mining of electronic health records for heart failure subtyping in 2023. Their paper was focused on evaluating whether text mining of electronic health record data can be used to improve register-based heart failure (HF) subtyping or not. The EHR data of 43,405 individuals were extracted from two Finnish hospital biobanks for mentioning the unstructured text of the Eruption Fraction (EF), and two 100-subject groups were randomly chosen versus the clinical evaluation. The structured laboratory data were then included for classification based on the HF subtype (Vuori, M. A., Kiiskinen, T., et al., 2023).

Amirhossein Jalilzadeh Afshari, M.S.S. (2018) proposed a model to predict the condition of patients according to their electronic records using time elements based on combining the artificial bee colony (ABC) algorithm and support vector machine. According to them, the issue of predicting diseases is one of the most critical issues today in the healthcare field, which is highly important. The physician makes this prediction based on the patient's clinical situation. Their research concluded that the best system for accurately diagnosing the disease can be developed based on the electronic file of the patient's conditions. They accomplished the prediction of the patient's condition according to their electronic files using time elements based on the combination of the artificial bee colony (ABC) algorithm and support vector machine. In their procedure, the artificial bee colony (ABC) algorithm was used for preprocessing and feature selection, followed by applying the decision support vector for classification. In the central part, two datasets were used to simulate

the proposed method to ensure its efficiency. The proposed method was compared with similar methods based on this simulation. Accordingly, the proposed method recorded a more appropriate improvement than those approached within the mentioned datasets. Their proposed method recorded 4% and 0.1% improvement rates in the heart and thyroid disease datasets, respectively [7].

Getzen et al. provided an exploratory model for equitable health in 2023 to assess the effect of missing data in electronic health records. According to them, electronic health records are gathered as a routine process of providing health care measures with a high potential to be used for improving patient health outcomes. These records contain multiple years of health data that can be used to predict risks, diagnose diseases, and evaluate treatments. However, they need a standardized and consistent format among institutions, especially in the United States, and can present significant analytical challenges. They encompass multi-scale data from heterogeneous domains and include structured and unstructured data. These data are gathered for individual patients at irregular intervals and with different frequencies. Besides analytical challenges, EHRs can reflect disparity; i.e., patients from different groups would have different amounts of data in their health records. Many of these issues can contribute to gathering biased data. As a result, the data from underserved groups may contain less information partly due to more sporadic care, which can be regarded as a missing data problem. There needs to be a framework for introducing missing values for the EHR data in this complicated form. Also, more research must be conducted to assess the effect of missing data in the EHR. In their work, they first introduced a term to define the three levels of EHR data. Then, they suggested a new framework to simulate real scenarios of missing data in the EHR to sufficiently evaluate their impact on predictive modeling. They combined a medical knowledge graph in the model to find and record dependencies between medical events to develop a more realistic missing data framework. They realized in the ICUs that missing data had a more significant negative impact on the performance of disease prediction models in groups tending to have less access to health care or seeking less health care. They also found that the effect of missing data on disease prediction models is more potent when using the knowledge graph framework for introducing actual missing values rather than eliminating random events (Getzen, E., Ungar, L., Mowery, D., Jiang, X., & Long, Q., 2023).

Mukherjee, P., Humbert-Droz, M., Chen, J. H., & Gevaert, O. (2023) proposed the SCOPE model to predict future diagnoses in office visits using electronic health records. They suggested an interpretable and scalable model to predict probable diagnoses in an encounter based on previous diagnoses and laboratory results. This model is designed to assist physicians in interacting with electronic health records. To do so, the EHR data of 2,701,522 patients at the Stanford Healthcare Center were collected and identified from January 2008 to December 2016. A population-based sample of 524,198 patients with multiple encounters with at least one recurrent diagnosis code was chosen. Then, a calibrated model was developed to predict ICD-10 diagnosis codes in an encounter based on previous diagnoses and laboratory results by utilizing a multi-label modeling

strategy based on binary communication. Logistic regression and random forests were examined as base classifiers, and multiple time windows were tested to gather previous diagnoses and laboratory results. This modeling approach was compared with the deep learning method based on the recurrent neural network. The best model utilized random forests as the base classifier and integrated demographic characteristics, diagnosis codes, and laboratory results. The best model had been calibrated, and its performance was comparable to or better than existing methods concerning different criteria, including the average AUROC of 0.904 in 583 diseases. The average AUROC with the best model was equal to 0.796 when predicting a patient's first occurrence of a disease label. The modeling approach performed better in terms of the AUROC ($p < 0.001$) compared to the tested deep learning method; however, it showed a lower performance regarding the AUPRC ($p < 0.001$). The interpretation of the model revealed that the model uses meaningful features and highlights many exciting relationships between diagnoses and laboratory results. The paper concluded that the multi-label model performs better than the RNN-based deep learning model, providing simplicity and potentially superior interpretability. While this model has been trained and validated on data obtained from a single institution, its simplicity, interpretability, and functionality make it a promising deployment candidate.

In 2023, Mohapatra et al. suggested a prediction model for heart diseases based on stacking classifiers. Cardiovascular diseases or heart diseases are known as one of the most substantial causes of death around the world. As estimated, about 1 out of every 4 deaths are caused by cardiovascular diseases, which are extensively classified as different types of abnormal heart diseases. However, diagnosing cardiovascular diseases is a time-consuming process in which the data obtained from different clinical trials are manually analyzed. Therefore, new approaches need to be developed to automate the detection of such abnormalities in human cardiac conditions to ultimately provide physicians with faster analysis by reducing diagnosis time and increasing results. Electronic health records are often utilized to discover valuable data patterns that help improve the prediction of machine-learning algorithms. In particular, machine learning helps solve problems like prediction in various domains, such as healthcare. Given the abundance of available clinical data, it seems necessary to use such information for the betterment of humanity. To this end, they presented a prediction model to predict heart diseases based on stacking different classifiers in two levels (basic and meta-level). Various heterogeneous learners are combined to generate robust model results. This model achieved a 92% precision rate in prediction with a 92.6% accuracy score, 92.6% sensitivity, and 91% specificity. The model performance was evaluated using different criteria, including accuracy, precision, recall, F1 scores, and area under the ROC curve values (Mohapatra, S., Maneesha, S., Patra, P. K., & Mohanty, S., 2023).

Essam et al. presented a model in 2023 for identifying heart disease risk factors according to electronic health records using advanced NLP and deep learning techniques. According to them, heart disease has still saved its place as the leading cause of death of people despite recent advances in the context of prediction and prevention. Thus,

identifying risk factors is a significant step in diagnosing and preventing heart disease. Automated detection of heart disease risk factors in clinical notes can contribute to modeling disease progression and clinical decision-making. Many studies have attempted to identify risk factors for heart diseases; however, none have identified all risk factors. These studies have proposed hybrid systems that combine knowledge-based and data-driven techniques based on dictionaries, rules, and methods of machine learning, which require significant human efforts. The National Informatics for Integrating Biology and the Bedside (i2b2) proposed a clinical natural language processing (NLP) challenge in 2014 with a track (Track2) focused on identifying risk factors for cardiovascular diseases risk factors in clinical notes over time. Clinical narratives provide a wealth of information that can be extracted using NLP and deep learning techniques. This paper was designed to improve the previous work in this field as a part of the i2b2 2014 challenge by identifying tags and features related to disease diagnosis, risk factors, and drugs by presenting advanced techniques of using stacked word embeddings. The i2b2 heart disease risk factors challenge dataset obtained using the stacked embeddings approach, which combines different embeddings, has remarkably improved. The model of this paper achieved an F1 score of 93.66% by utilizing BERT and character embedding (CHARACTER-BERT Embedding). The proposed model benefits from substantial results compared to other models and systems we have developed for the i2b2 2014 challenge (Houssein, E. H., Mohamed, R. E., & Ali, A. A., 2023).

Liang et al. suggested a disease prediction model based on integrating the data of several types of China's electronic health records. They state that disease prediction using various healthcare data to help doctors diagnose diseases has recently become a more prominent research topic. Their paper proposed a disease prediction model that combines different types of encrypted representations of Chinese EHRs. The model framework uses a Multi-head self-attention mechanism that combines textual and numeric features to improve the text representation. The BiLSTM-CRF and TextCNN models are used to extract entities and obtain representations from them. Text representations and the entities contained therein are combined to formulate the representations of electronic health records. The experimental results on electronic health record data gathered from a Class B general hospital in Gansu Province, China, indicated that their model has an F1 score of 91.92, which is better than the previous basic methods (Liang, Z., Zhang, Z., Chen, H., & Zhang, Z., 2022).

In 2022, Rakhmetulayeva, S., & Kulbayeva, A. (2022) presented a model for disease prediction using machine-learning algorithms based on electronic health record reports. They believed that the number of tasks assigned to predict the occurrence of infectious diseases is on a rapid growth path because of the availability of statistical data that supports the relevant analysis. Their paper described the current leading solutions to make short-term and long-term disease predictions. Also, the limitations and practical applications of these solutions have been mentioned. The paper gave much attention to the Naive Bayes classification, logistic regression, artificial neural network algorithm, and

k-means artificial neural networks as model analysis methods based on machine learning. The article provided an overview of two popular machine-learning algorithms for disease prediction. It used standard datasets for various diseases, including fungal infections, allergies, GERD, chronic cholestasis, stomach ulcer disease, diabetes, bronchial asthma, migraine, paralysis (cerebral hemorrhage), etc.

Zhao et al. provided a disease progression prediction model based on data from electronic health records in 2022. They suggested that electronic health records encompass patients' diagnostic, hospitalization, and medication records, in which a tremendous amount of structured time series data is at reach. Significant advances have occurred in electronic health record analysis and mortality prediction research. However, available electronic health records data have a sporadic and irregular nature, which prevents the accomplishment of scientific research and practical applications based on time-series electronic health records data. This paper generated several models based on deep neural networks to evaluate the prediction of patients' mortality. First, an attention mechanism was introduced to extend the factoring machine model, which dynamically learns the weights of different feature combinations to obtain some interpretability in the model. Second, the bidirectional gated regression unit (BiGRU) was applied to simultaneously capture the long-term dependencies in the forward and backward directions. Third, the BiGRU-AFM model was proposed and extensively examined in data mining based on electronic health records. The principal, second-order, and higher-order features are utilized to achieve a complete combination of features and a comprehensive emotional interaction in electronic health records. In particular, the attention-based FM (AFM) part presupposes combinations of low-order features, and the BiGRU section records higher-order feature interaction vectors. Their joint output appeared to make highly expressive vectors to predict the patients' mortality. Finally, a series of experiments were conducted on the public electronic health record dataset, in which experimental results demonstrated that the proposed BiLSTM-FM model performed better than the advanced basic models and gained about 97.9% in the widely used metric of the area under the curve (Zhao, F., Yu, X., Zhang, J., Li, X., & Li, R., 2022, December).

2. The analysis of the proposed method

Predicting the condition of patients in medicine and health has become highly important and essential. To achieve this goal, access to adequate information on the patient's complete health records is essential for accurate prediction. The first innovation to implement this section in the current research was using electronic health records (EHR). Utilizing this system provides sufficient information to the doctor or intelligent system. It has proven much better than the traditional manual systems available in many medical fields. A growing body of literature uses data obtained from EHRs to shape and design medical and health studies. Also, there is a growing but much smaller body of literature on essential and intrinsic data quality (DQ) problems in the EHRs as a source of research data, which is involved with the broad range of non-accidental human errors in different dimensions.

Some frameworks, such as those introduced by Weiskopf, N. G., & Weng, C. (2013) and Kahn, M. G., et al. (2016), may be used to classify the EHR DQ dimensions and help identify appropriate strategies for reduction. Selecting them appears essential in the EHR research. Process mining in healthcare can be challenging due to highly different care patterns among patients, healthcare professionals, and organizations. Thus, relying on this approach to complete those event reports with a time stamp can enhance measurable DQ requirements. Systematic event mapping, reconstruction, and analysis techniques are essential, just like other forms of data mining, because they are necessary for transparency about data cleaning and control procedures. DQ problems may happen during the life cycle of EHR data, from the design time of the original EHR application and database and its application in practice to data extraction for research, technologies, and methods used in them.

Another innovation of this research came from using the evolutionary shuffled frog leaping algorithm (SFLA) for "feature selection" to reduce the dimension and improve the quality of the available data. Also, a standard classification was needed in the final part of the current research for an optimal outcome to predict the patient's health status, which was met by the deep learning technique in the proposed system. The block diagram of the proposed method is illustrated in Figure 1.

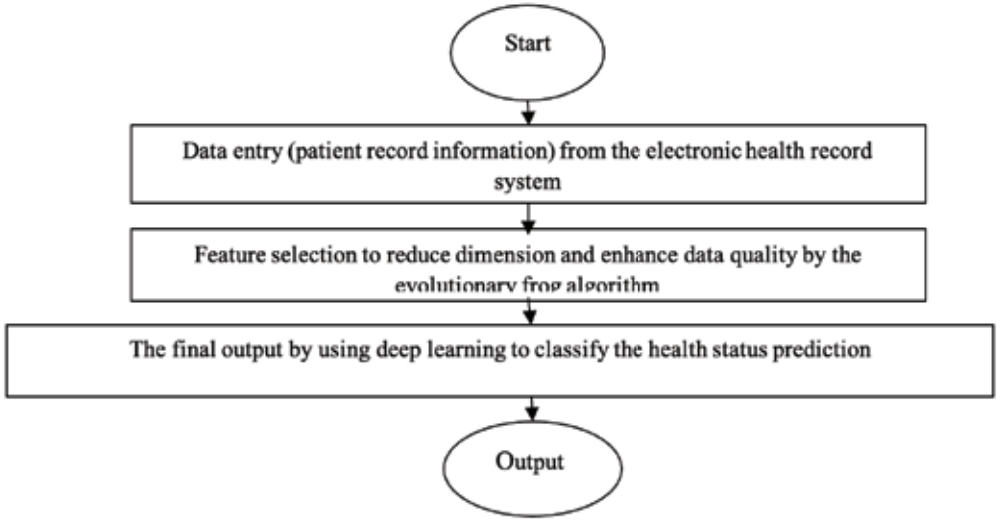


Fig.1: The block diagram of the proposed method

The general approach of the proposed method consists of the following steps (Weiskopf, N. G., & Weng, C., 2013):

o Preprocessing and selection of proper features

o We are identifying the DQ dimensions, detecting possible sources of information

related to the DQ, preparing a list of potential DQ problems, establishing links between these problems and experiments, data marking, and reducing the DQ problem if possible.

o Studying: The analysis of the results of the "Do" phase.

o Action: Taking measures to improve the future DQ.

The goal here is to specify data with non-acceptable quality as "bad" data, i.e., unusable. Incomplete but acceptable data are determined as "moderate" data, meaning that they can be utilized in some circumstances or experiments. Other data are unspecified or "good" and available for all purposes. The use of the proposed method consists of three major stages. The first stage involves identifying the archive of DQ problems for the research. The known problems are moved to this archive in advance. This archive will be completed with other issues specific to the research questions (the output of stage 1 includes the three dictionary entities: dimension, levels, and sources, as well as the very archive of the DQ issues and the list of tests for this research). The second stage presents the research data through analysis and classification tools. The third stage involves preparing a report about what has happened (Weiskopf, N. G., & Weng, C., 2013).

3. The analysis of simulation results of the proposed method

The simulation results of the proposed method were examined in this section, and the results of different evaluation stages were represented in Matlab and Weka software in the form of graphs and tables. Moreover, two datasets were utilized to assess the proposed method. The Heart Disease and Thyroid Disease datasets obtained from the UCI databases were used in the evaluations. The Heart Disease dataset included 303 samples with 75 features for each sample. Due to using electronic health records, the features are in two different sections, including patients' personal information and clinical and disease information. The dataset had missing data (values), just like the real-world datasets. The thyroid Disease dataset included 7200 samples in 10 different sets with 21 features, of which a set with 2800 samples was used in the evaluation. This set also had missing data, and the so-called used sets were not clean. The profile tables of the two datasets are shown in Figure 2.

Data Set Characteristics:	Multivariate	**Number of Instances:**	303	**Area:**	Life
Attribute Characteristics:	Categorical, Integer, Real	**Number of Attributes:**	75	**Date Donated**	1988-07-01
Associated Tasks:	Classification	**Missing Values?**	Yes	**Number of Web Hits:**	981302

A. The Heart Disease dataset

Data Set Characteristics:	Multivariate, Domain-Theory	**Number of Instances:**	7200	**Area:**	Life
Attribute Characteristics:	Categorical, Real	**Number of Attributes:**	21	**Date Donated**	1987-01-01
Associated Tasks:	Classification	**Missing Values?**	N/A	**Number of Web Hits:**	192054

B. The Thyroid Disease dataset

Fig. 2: The specifications of the two datasets

4. The parameters used to select the feature of the proposed method

The shuffled frog leaping algorithm (SFLA) was used in the proposed method to improve the process of discovering similarities and help the final clustering in extracting the impact of nodes and the proximity of influential factors to each other and weighting these factors as well as to select proper parameters for determining the most critical parameters, reducing the conclusion time, and enhancing the quality. The parameters considered for this algorithm are given in Table 1.

Table 1: The parameters used in the shuffled frog leaping algorithm

Row	Parameter Name	Description	Value
1	nVar	Number of decision-making variables	Equal to the number of features
2	VarMin	The lower limit of the variable values	-10
3	VarMax	The higher limit of the variable values	10
4	MaxIt	Maximum iteration of the algorithm	300
5	nPop	The number of frogs	50
6	nPopMemeplex	Memeplex size	10
7	nMemeplex	Number of Memeplex	5
8	alpha	Alpha value	3
9	beta	Beta value	5
10	sigma	Sigma value	2

The fixed values used in the algorithm have been obtained from repeated iterations and represent the best output with these values. An instance of the result of this algorithm is depicted in Figure 3.

As illustrated in Figure 3, the value of the best cost is high at the beginning of implementing the algorithm, and the algorithm tries to minimize and incline it towards zero, which approaches this value by advancing in the number of iterations. However, choosing the maximum number of iterations is essential in evolutionary and optimization algorithms such as the frog leaping algorithm since a small number may result in an improper best cost, and a large number may enhance the response time. Thus, choosing the correct value seems very important, and usually, this value is obtained by repeating the execution and observing the result. The algorithm has been implemented with different iterations in the proposed method aimed at achieving optimal results. The results are shown in Figure 4.

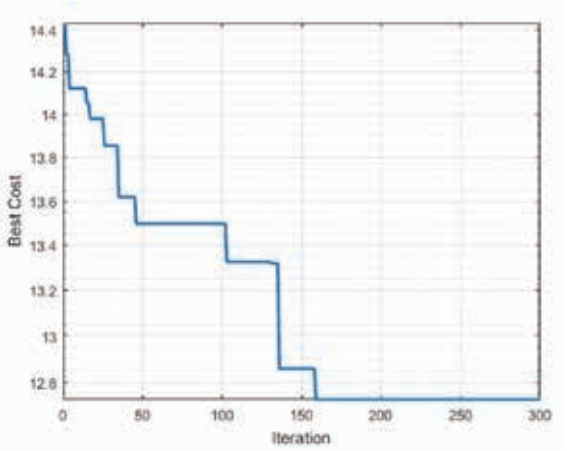


Fig. 3: The result of implementing the shuffled frog leaping algorithm

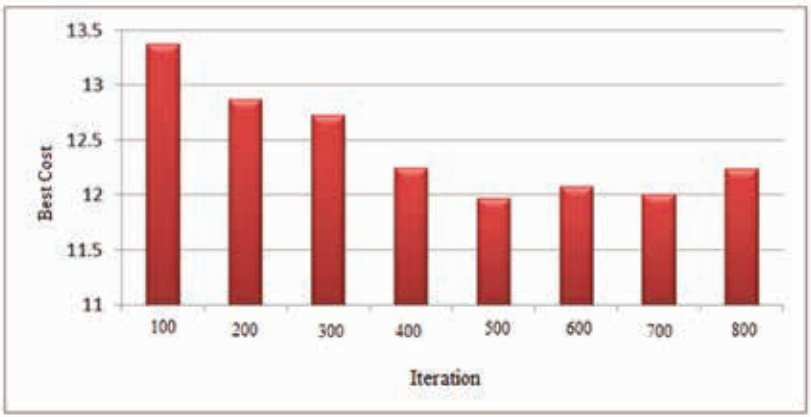


Fig. 4: Comparing the value of the best cost in different implementations of the frog leaping algorithm

As illustrated in Figure 4, the results in different implementations and the best cost value follow a decreasing trend. This reduction continued until repetition 500, and the repetitions increased to 100 units at each stage. This decrease in value has been slight in rounds 400 and 500 and increased after that. Hence, the value of 500 has been used as the maximum number of iterations of the frog leaping algorithm in the proposed method.

There are three different cost functions to choose from according to the problem in the frog leaping algorithm. These three functions are given in Table 2.

Table 2: Different functions to be used in the frog leaping algorithm

Row	Function Name	Calculation procedure

1	Rosenbrock	z=sum((1-x(1:n-1)).^2)+100*sum((x(2:n)-x(1:n-1).^2).^2);
2	Ackley	z=20*(1-exp(-0.2*sqrt(mean(x.^2))))+exp(1)-exp(mean(cos(2*pi*x)));
3	Sphere	z=sum(x.^2);

As shown in Table 2, there are three functions to choose for calculating the best cost in the algorithm. Accordingly, the algorithm was implemented with identical conditions and 300 repetitions with all three functions, and the results of the best final cost are provided in Figure 5.

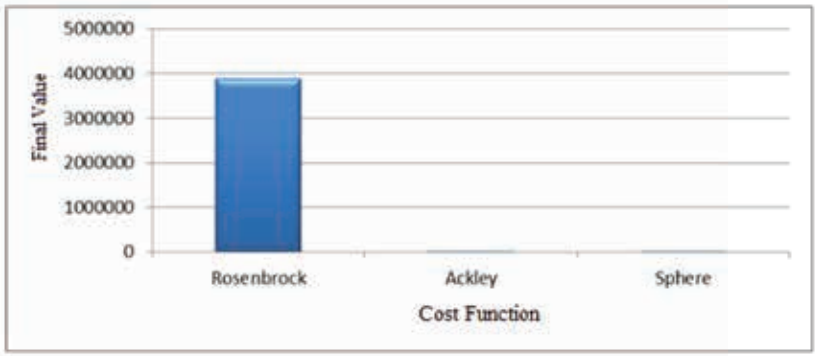


Fig. 5: The comparison of the functions that can be used in the frog leaping algorithm in terms of the value of the best final cost

As shown in Figure 5, the Rosenbrock and the Ackley functions have obtained the worst and best values at the end of the execution and after 300 iterations, respectively. The diagram of the cost function value change in the three compared functions is illustrated in Figure 6.

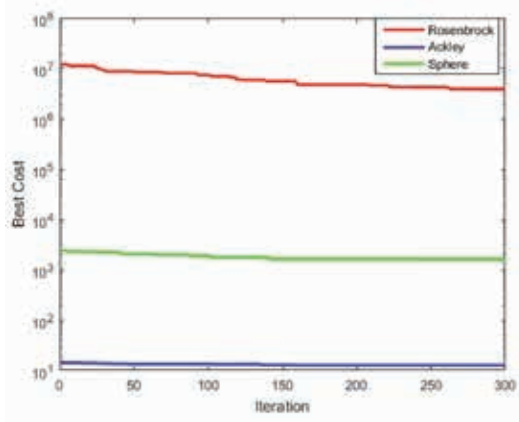


Fig. 6: The comparison of the value of the cost function at different iterations of the frog leaping algorithm with three usable functions

As shown in Figure 6, the Rosenbrock and the Ackley functions have shown the worst and best values at the end of the execution and after 300 iterations, respectively. According to these results, the Ackley function was used in the route selection section of the proposed method. The dataset following feature selection by adding the SF is shown below.

5. Comparing the proposed method with similar methods

The obtained collection was then evaluated and compared with different classification methods to determine the strength and improvement of the proposed method compared to the known and widely used methods. The criteria used for the evaluation are given in Table 3. The validation test by the K-Fold method with K=10 was used in all experiments. In this type of validation, the data is divided into K subsets. One of these K subsets is used for validation each time, and the other K-1 will be used for training. The K-Nearest Neighbor classification algorithms, J48, SMO, Decision Table, and the algorithm based on the Bayes theory were employed for evaluation.

Table 3: Introducing the evaluated criteria in the proposed method

Criterion	Description
Accuracy	The closeness of the agreement between the average value obtained from many test results and the accepted reference value is also called "average accuracy."
TP Rate	It represents the number of records whose real category is positive, and the classification algorithm has also recognized their category as positive.
FP Rate	It represents the number of records whose actual category is negative, and the classification algorithm has mistakenly recognized their category as positive.
Recall	A general parameter is used to evaluate the usefulness of the proposed algorithm and acts as the following equation (Th is the set of members inside each class, and Tr is the set of members inside the algorithm). $Recell= (T_h \cap T_r)/T_h$
Precision	A general parameter is used to measure the usefulness of the proposed algorithm and is obtained from the following relation. $Precision= (T_h \cap T_r)/T_r$
F-Measure	This criterion is obtained by calculating the average of the correlation between the two criteria of usefulness and utility, which is obtained by using Recall(R) and Precision(P) parameters in the form of the following equation. $F=2PR/P+R=2/(1/R+1/P)$

6. The first evaluation of the validity criteria results on the thyroid disease and heart disease datasets

As seen in Figure 7, in the heart disease dataset, followed by applying the DQ proposed in the previous chapter, all methods have demonstrated more acceptable and favorable results compared to the same set before this DQ application. Before applying the data quality improvement on the dataset, the support vector classification method did not have a good performance, and 67% of accuracy was in the third category of compared algorithms, or the proposed deep learning method with 89.3% has been in a place two algorithms above the rule Bayes algorithm method. After applying the proposed method, obtaining a new dataset, and repeating the experiments on this dataset, the recorded results indicated performance improvement and optimization. The deep learning method ranked first with an accuracy of 95.8%. It is worth noting that the results of all evaluated algorithms improved after applying the proposed DQ in the previous chapter. Figure 7 shows the results of the accuracy evaluation of these methods on the thyroid disease dataset.

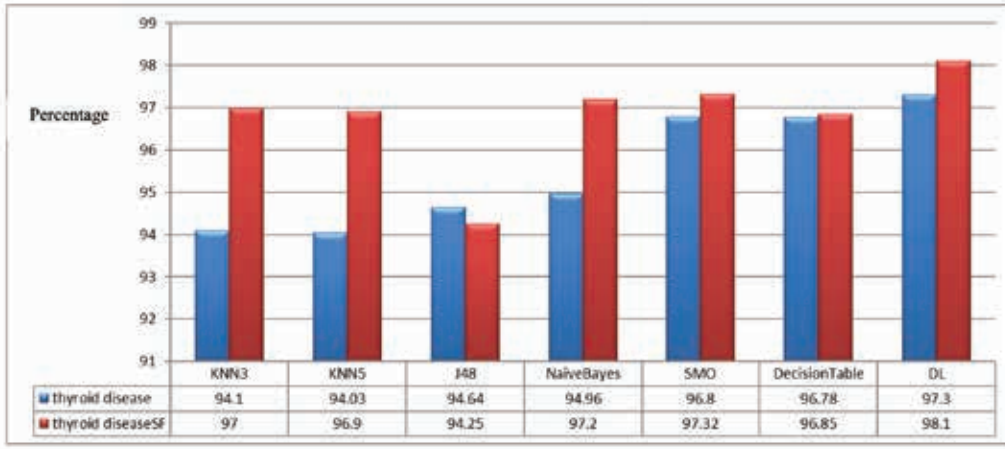
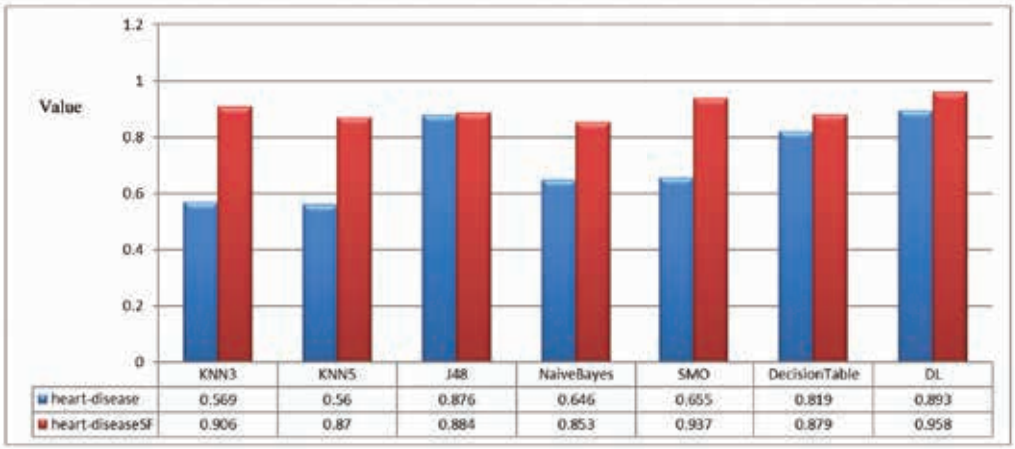


Fig. 7: Comparing the accuracy of the proposed method with other methods in the dataset of thyroid diseases

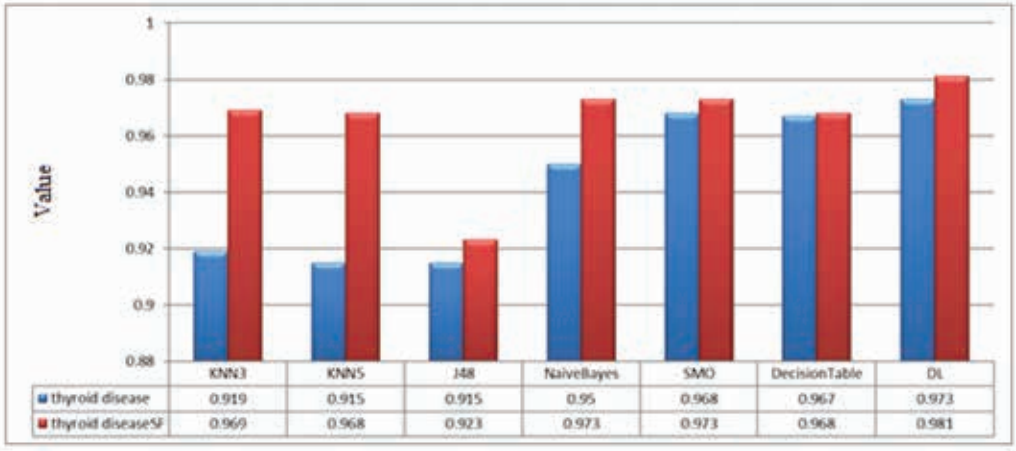
As shown in Figure 7, the evaluated methods have had a lower performance than the proposed method. However, it should be noted that the results of the Bayesian method are very close to those of the proposed method. After applying the DQ steps to the data, all methods have shown improvement and optimization. The deep learning method has performed better than other methods before and after applying the DQ in this dataset. The proposed method and other similar methods were evaluated on two datasets with the F-Measure to enhance the reliability of the evaluation process. The results of this evaluation of the heart disease dataset are illustrated in Figure 8.



	KNN3	KNN5	J48	NaiveBayes	SMO	DecisionTable	DL
heart-disease	0.569	0.56	0.876	0.646	0.655	0.819	0.893
heart-diseaseSF	0.906	0.87	0.884	0.853	0.937	0.879	0.958

Fig. 8: The comparison of the F-Measure of the proposed method and other methods in the heart disease dataset

As shown in Figure 8, all methods have demonstrated more acceptable and favorable results after applying the DQ proposed in the previous chapter compared to the same set before applying these measures in the heart disease dataset. Before applying the data quality improvement on the dataset, the proposed classification method had an F-Measure value of 0.893 and has been two algorithms above the rule Bayes algorithm method. After applying the proposed method, obtaining a new dataset, and repeating the tests on this dataset, the recorded results indicated performance improvement and optimization. The deep learning method ranked first with an F-measure value of 0.958. It is worth noting that the results of all the evaluated algorithms improved after applying the proposed DQ in the previous chapter. Figure 9 shows the results of the F-Measure evaluation of the methods on the thyroid disease dataset.



	KNN3	KNN5	J48	NaiveBayes	SMO	DecisionTable	DL
thyroid disease	0.919	0.915	0.915	0.95	0.968	0.967	0.973
thyroid diseaseSF	0.969	0.968	0.923	0.973	0.973	0.968	0.981

Fig. 9: The comparison of the F-Measure of the proposed method and other methods in the thyroid disease dataset

As shown in Figure 9, the evaluated methods had a lower performance than the proposed method. However, it should be noted that the results of the Bayesian method were very close to the proposed method. All methods demonstrated improvement and optimization after applying the DQ steps of the data. The deep learning decision method has performed better than other methods before and after applying the DQ in this dataset. The proposed method was compared with the results of papers (Mohapatra, S., Maneesha, S., Patra, P. K., & Mohanty, S., 2023) and [7] in the final evaluation. In the article [7], the author has performed the proposed method with the Artificial Bee Colony (ABC) and Decision Support Vector Machine (DSVM) algorithms. In contrast, the article (Mohapatra, S., Maneesha, S., Patra, P. K., & Mohanty, S., 2023) has only worked on the heart dataset, and this evaluation has been made merely on this set. The evaluation of the proposed method is depicted in Figure 10.

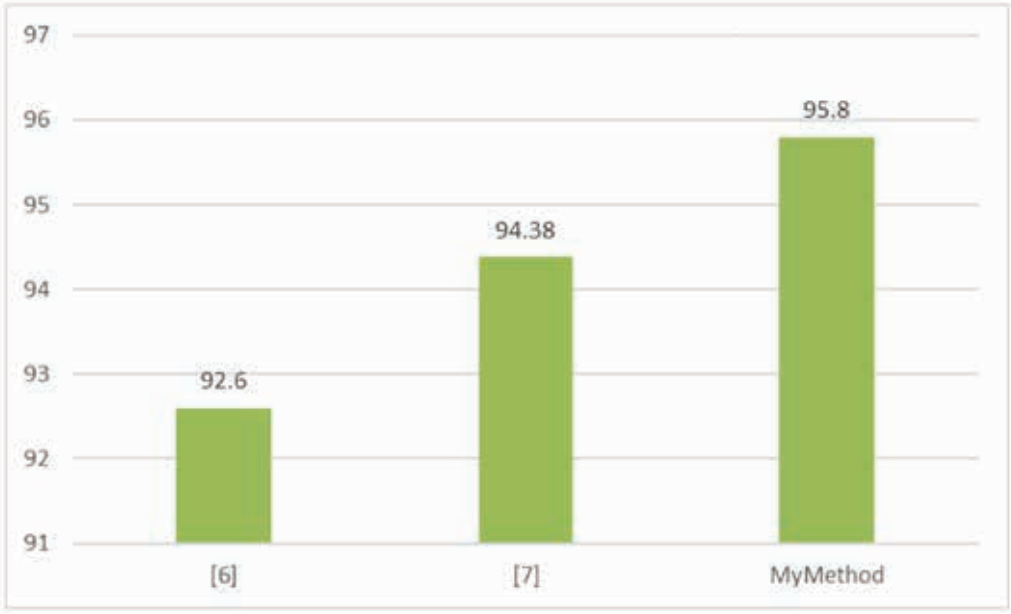


Fig. 10: The evaluation of the proposed method and similar papers in terms of accuracy criterion

As shown in Figure 10, the proposed method recorded an accuracy value of 95.8%, which has recorded an improvement of 1.4 compared to the same method of the present author, with an accuracy of 94.38%. Also, it shows an improvement of 3.2% compared to the similar work of the paper (Mohapatra, S., Maneesha, S., Patra, P. K., & Mohanty, S., 2023), which has a value of 92.6%.

Conclusion and summary

This research provided a method to create a proper platform for analyzing and using time data, which was used to analyze the data, followed by making the necessary

changes to advance and enhance the data quality based on the analysis. Accordingly, the goal was set to enhance the quality and optimize the data in future measures. The evolutionary shuffled frog leaping algorithm (SFLA) was used for preprocessing and feature selection in the proposed method, followed by the deep learning technique for classification. Accordingly, all the requested results and information were examined, and then the method was simulated. Two datasets were used in the simulation to ensure the method's performance. Based on this simulation, the proposed method improved the evaluated datasets compared to similar methods. The improvement rates were recorded as 1.4% and 3.2% in the heart disease dataset compared to the author's previous and updated methods, respectively.

For future work, there is scope to manage an application for collaboration of institutions that acquire and process such data and can perform clinical-level ML computations to solve real-time problems. This program leads to faster decision-making. The same method can be used to diagnose other chronic diseases as well. Implementing the proposed method in the real world can improve performance, including future work on the proposed method aimed at finding and fixing its shortcomings. Moreover, using other optimization and evolutionary algorithms rather than the frog optimization algorithm may bring different results, which need to be evaluated and suggested if they would be better.

References

Amirhossein Jalilzadeh Afshari, M.S.S. (2018). Predicting the condition of patients from electronic records using temporal elements based on the combination of bee colony algorithm and support vector machine. Master thesis, Azad University

Getzen, E., Ungar, L., Mowery, D., Jiang, X., & Long, Q. (2023). Mining for equitable health: Assessing the impact of missing data in electronic health records. *Journal of Biomedical Informatics, 139,* 104269.

Hossain, M. E., Khan, A., Moni, M. A., & Uddin, S. (2019). Use of electronic health data for disease prediction: A comprehensive literature review. *IEEE/ACM transactions on computational biology and bioinformatics, 18*(2), 745-758.

Houssein, E. H., Mohamed, R. E., & Ali, A. A. (2023). Heart disease risk factors detection from electronic health records using advanced NLP and deep learning techniques. *Scientific Reports, 13*(1), 7173.

Kahn, M. G., et al. (2016). A harmonized data quality assessment terminology and framework for the secondary use of electronic health record data. *Egems, 4*(1).

Liang, Z., Zhang, Z., Chen, H., & Zhang, Z. (2022). Disease prediction based on multi-type data fusion from Chinese electronic health record. *Math. Biosci. Eng, 19*(12), 13732-13746.

Lu, H., & Uddin, S. (2023, April). Disease Prediction Using Graph Machine Learning Based on Electronic Health Data: A Review of Approaches and Trends. In *Healthcare* (Vol. 11, No. 7, p. 1031). MDPI.

Mahmoudi, E., Kamdar, N., et al. (2020). Use of electronic medical records in development and validation of risk prediction models of hospital readmission: systematic review. *bmj, 369*.

Mohapatra, S., Maneesha, S., Patra, P. K., & Mohanty, S. (2023). Heart Diseases Prediction based on Stacking Classifiers Model. *Procedia Computer Science, 218,* 1621-1630.

Mukherjee, P., Humbert-Droz, M., Chen, J. H., & Gevaert, O. (2023). SCOPE: predicting future diagnoses in office visits using electronic health records. *Scientific Reports, 13*(1), 11005.

Mulla, F. D., & Jayakumar, N. (2018, November). A Review of Data Mining & Machine Learning approaches for identifying Risk Factor contributing to likelihood of cardiovascular diseases. In *2018 3rd International Conference on Inventive Computation Technologies (ICICT)* (pp. 631-635). IEEE.

Rakhmetulayeva, S., & Kulbayeva, A. (2022). Building Disease Prediction Model Using Machine Learning Algorithms on Electronic Health Records' Logs. DTESI.

Vuori, M. A., Kiiskinen, T., et al. (2023). Use of electronic health record data mining for heart failure subtyping. *BMC Research Notes, 16*(1), 208.7.

Weiskopf, N. G., & Weng, C. (2013). Methods and dimensions of electronic health record data quality assessment: enabling reuse for clinical research. *Journal of the American Medical Informatics Association, 20*(1), 144-151.

Woldemariam, M. T., & Jimma, W. (2023). Adoption of electronic health record systems to enhance the quality of healthcare in low-income countries: a systematic review. *BMJ Health & Care Informatics, 30*(1).

Zhao, F., Yu, X., Zhang, J., Li, X., & Li, R. (2022, December). A Disease Progression Prediction Model Based on EHR data. In *2022 IEEE 24th Int Conf on High Performance Computing & Communications; 8th Int Conf on Data Science & Systems; 20th Int Conf on Smart City; 8th Int Conf on Dependability in Sensor, Cloud & Big Data Systems & Application (HPCC/DSS/SmartCity/DependSys)* (pp. 1625-1632). IEEE.

Submitted 12.09.2023
Accepted 02.11.2023

Azerbaijan Journal of High Performance Computing, Vol 6, Issue 2, 2023, pp. 153-162
https://doi.org/10.32010/26166127.2023.6.2.153.162

Application of AHP for Weighting Clients in Federated Learning

Samir Aliyev

Azerbaijan State Oil and Industry University, Baku, Azerbaijan, aliyev.samir@asoiu.edu.az

*Correspondence: Samir Aliyev, Azerbaijan State Oil and Industry University, Baku, Azerbaijan, aliyev.samir@asoiu.edu.az

Abstract

Federated Learning is a branch of Machine Learning. The main idea behind it, unlike traditional Machine Learning, is that it does not require data from the clients to create a global model, so clients keep their data private. Instead, clients train their model on their own devices and send their local model to the server, where the global model is aggregated and sent back to clients. In this research work, the Federated Averaging algorithm is modified so that clients get their weights by the Analytical Hierarchal Process. Results showed that applying AHP for weighting performed better than giving clients weights solely based on their dataset size, which the Federated Averaging algorithm does.

Keyword: Federated Learning, AHP, Geometric Mean, Client Weighting, Federated Averaging.

1. Introduction

Federated Learning is a newly emerging machine learning technique that emphasizes decentralized model training and privacy. Sensitive data is kept local and is not sent to a central server in this collaborative paradigm because models are trained across several decentralized devices or servers. Through iterative learning, each participating device improves its local model; only the updated models are transmitted to the central server. This allows for ongoing development without jeopardizing the privacy of personal information. Federated learning is beneficial when data is dispersed among multiple sources, including IoT, edge, and mobile devices. Applications in healthcare, finance, and other industries where data privacy is a concern benefit greatly from its ability to increase efficiency and decrease the need for large-scale data transfers. One of its main advantages is its ability to handle non-IID (non-Independently and Identically Distributed) data distributions, which addresses real-world scenarios where data characteristics may vary considerably.

Aggregation is a crucial phase in the cooperative model training process in federated learning. Once individual servers or devices have trained on their respective datasets to create local model updates, these updates are combined at a central server to create the aggregated model. Calculating the weighted average of the model parameters across all participating devices is a common step in the aggregation process. Depending on the particular needs of the FL system, various aggregation techniques, such as simple or weighted averaging, can be used. One of these methods is Federated Averaging

(FedAVG). In the FedAVG model, parameters are weighted and averaged. Weighting is respective to the size of the dataset clients used for training. The main goal of this work is to propose different methods to give weights to clients using other facts such as computing capabilities and distribution of classes in the clients using expert systems.

Section 2 is dedicated to related research on model aggregation and client weighting. Section 3's central methodology is about how our approach works for aggregation in FedAVG using the Analytical Hierarchal Process (AHP). The results, comparisons, and training curves are demonstrated in section 4.

2. Literature Review

Several works have been devoted to aggregating models' weights in federated learning. Federated averaging is one of the earliest and most cost-effective methods used in federated learning (T. Sun, D. Li and B. Wang, 2023). However, it is a very naive approach that only uses the training set size to give the weights to the client. In this case, clients with small data set sizes can hardly influence the model. Considering that some clients may have "healthier" data even at small sizes, assigning more weights can potentially improve the performance of the global model (Li, Y., Guo, Y., Alazab, M., Chen, S., Shen, C., & Yu, K., 2022). To overcome this difficulty, several works have been carried out (Qi, P., Chiaro, D., Guzzo, A., Ianni, M., Fortino, G., & Piccialli, F., 2023).

One such work is done by applying forgetting (Xu, C., Hong, Z., Huang, M., & Jiang, T., 2022). Forgetting in Federated means that one observation is classified correctly on the local model, but the global model should have classified that observation. In this case, the new models will likely forget previous observations because the global model aggregates the local model. Because of this local model, performance decreases when new batches are tested. To overcome this difficulty, Federated Weighted Averaging (FedWAvg) was proposed (Hong, M., Kang, S. K., & Lee, J. H., 2022). It gives clients weights based on the local forgettable examples. The clients with more forgotten examples. This rebalances the global model and makes clients with forgetful examples less affected by global updates. Experiments showed that the proposed approach performed better than the previous algorithm.

In work (Ye, R., Xu, M., Wang, J., Xu, C., Chen, S., & Wang, Y., 2023), the authors considered local and global distribution through the FedDisco algorithm. The results showed that there are better ways than just using the size of the dataset to weight clients. FedDisco algorithm outperformed the FedAvg accuracy score by about 17%. In another work (Tang, Z., Shao, F., Chen, L., Ye, Y., Wu, C., & Xiao, J., 2021), weights are assigned to clients dynamically. Instead of giving them static weights, the authors proposed a method that assigns weights based on the contribution in each round. The experiments were carried out in CIFAR-10 and MNIST. The proposed approach increased performance. The disadvantage of this approach is that the weights must be calculated in each round of contributions.

Expert systems were also used to assign weights to the customer. One of these works

was carried out to overcome the problem of non-IID data (Wilbik, A., & Grefen, P., 2021). In another work, the authors used various criteria, such as computing capacity and network resources, to give weights to clients (Du, Z., Wu, C., Yoshinage, T., Zhong, L., & Ji, Y., 2022). Fuzzy Inference Systems carried it out. In order to test the different calculation capabilities, different devices were used for training. The FIS system was also applied in federated learning in work by (Aliyev, S. & Ismayilova; N., 2023). This work also took into account the distribution of classes for each customer. Computational capacity, class distribution, and dataset size were passed to the Mamdani-type FIS as input, and the clients' weights were the output. The results showed that applying these criteria increased the performance of the global model. This research work is dedicated to the application of the Analytical Hierarchal Process for weighting the clients.

3. Methodology

3.1. AHP

Analytical Hierarchal Process is a multicriteria decision making method developed by Saaty (Saaty, R. W., 1987). In AHP, decision-makers must determine and understand the problem and its goal. Factors that can influence the decisions are evaluated and compared pairwise. These comparisons are given some value based on the importance of one factor over the importance of another. These values are evaluated by experts based on the state-of-art results, problems, experience from previous cases, and learning. The importance of the factor is taken as a value in the range of 1 and 9. One such scale adopted by Saaty is shown in Table 1 (Saaty, R. W., 1987).

Table 1. Importance scale

Value	Meaning	Description
1	Equal	Two factors influence the decision equally
3	Somewhat more important	One factor is slightly more important than the other for decision
5	Much more important	One factor is strongly preferred over the other in order to make a decision
7	Very much more important	One factor is much more preferred than the other. Its importance is already observed in practice
9	Absolute importance	The importance of one factor over another is evaluated in the highest possible validity.
2,4,6,8	Intermediate values	When compromises are needed.

Each value of the matrix is given value as:

$$A_{ij} = \frac{1}{A_{ji}}$$

AHP usually consists of three steps.

Decomposition: This step decomposes the problem in hierarchal form. Objectives and criteria are detected. If the hierarchy goes deeper, sub-criterions are also taken into consideration.

Pairwise comparison: In this step, the experts create a pairwise comparison matrix using Saaty's 9-point scale. Giving values to each comparison is the central concept in AHP. The priority vector is calculated using this matrix. Several methods for estimating the priority vector include geometric mean, fuzzy geometric mean, eigenvector method, etc.

Composition of priorities: In this step, priorities, starting from the lowest level to the highest level, are synthesized. The goal is to obtain overall priority that reflects the importance of each alternative.

3.2. Geometric Mean

Geometric Mean is one of the prioritization methods of AHP (Yadav, A., & Jayswal, S. C., 2013). The steps of this prioritization method are as follows:

Find the geometric mean of each row

$$GM_i = \left(\prod_{j=1}^{n} a_{ij} \right)^{\frac{1}{n}} \quad i = 0,1, \dots, n$$

Sum them up

$$The\ sum\ of\ Rows = \sum_{i=1}^{n} GM_i$$

Normalize each row by dividing by the sum of rows to obtain the priority vector

$$Vector_i = \frac{GM_i}{Sum\ of\ rows}$$

Calculate the CR:

$$CI = \frac{\lambda_{max} - n}{n - 1}$$

$$CR = \frac{CI}{RCI}$$

Where λ_{max} is the maximum eigenvalue of the matrix, and RCI is provided by:

If CR is less than 0.1, then the matrix is acceptable. Note that if CR is greater than 0.1, it does not necessarily mean the matrix is inconsistent.

Table 2 RCI table

N	RCI
1	0
2	0
3	0.5799
4	0.9
5	1.12
6	1.25

7	1.33
8	1.39

3.3. Steps

The following preference matrix used in this work is described in Figure 1:
Geometric means of each row:

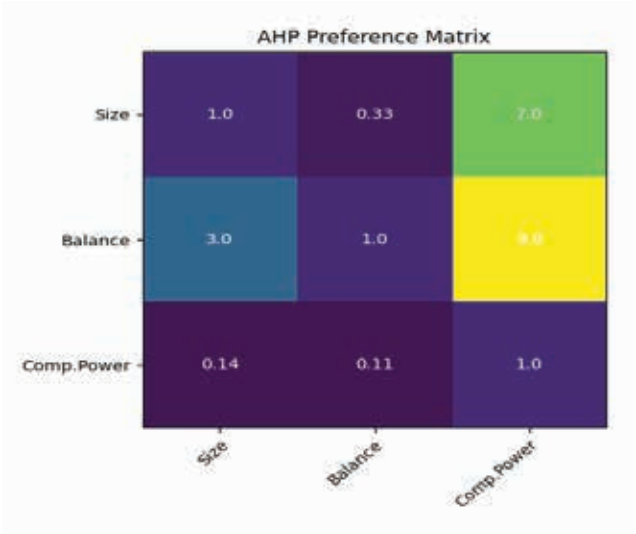


Fig. 1: Preference matrix used in this work

$$\sqrt[3]{1 * 0.33 * 7} = 1.32$$
$$\sqrt[3]{3 * 1 * 9} = 3$$
$$\sqrt[3]{0.14 * 0.11 * 1} = 0.25$$

The sum of them: 1.32+3+0.25 = 4.57

Priority vector is: $[\frac{1.32}{4.57}, \frac{3}{4.57}, \frac{0.25}{4.57}] = [0.29,\ 0.66, 0.05]$

Maximum eigenvalue is 3.08. Then CR is:

$$CI = \frac{3.08 - 3}{3 - 1} = 0.04$$
$$CR = \frac{0.04}{0.5799} = 0.068$$

CR is less than 0.1, which implies the matrix is consistent and can be applied for decision making.

3.4. Logistic Regression

Logistic Regression is a parametric linear machine learning algorithm used for classification tasks. It uses the sigmoid function to predict (Roberts, G., Rao, N. K., & Kumar, S., 1987). Sigmoid is an "S" shaped function. The threshold value controls the classification boundary. Because the distribution of classes among clients is not known in federated learning, 0.5 was used as a threshold for all clients. Sigmoid function can be defined as:

$$\sigma(x) = \frac{1}{1 + e^{-z}}$$

where z is:

$$z = \beta_0 + \sum_{i=1}^{k} \beta_i x_i$$

Where vector x is input and vector β are the model parameters.

The binary cross entropy function was used as a loss function and optimized by gradient descent algorithm.

3.5. Federated Averaging

Federated Averaging is one of the original approaches for aggregating model parameters. Federated averaging can be defined as:

$$For\ each\ client\ k, w_{i+1}^{k} \leftarrow w_i^k - \alpha g_i$$

$$w_{i+1} \leftarrow \sum_{i=1}^{k} \frac{n_k}{n} w_{i+1}^{k}$$

where α is learning rate, g_i is gradient in i^{th} iteration for each client. n_k is the size of a training set of client k, and w_{i+1}^{k} is the updated weights in iteration i+1 for each client k.

The parameters of the model can be updated for several iterations before sending it to the server, reducing communication costs and making parallelization easier.

3.6. Our Approach

Our approach changes the FedAVG algorithm so clients get weights based on AHP results. So the overall formula is defined as:

$$For\ each\ client\ k, w_{i+1}^{k} \leftarrow w_i^k - \alpha g_i$$

$$w_{i+1} \leftarrow \sum_{i=1}^{k} A_k w_{i+1}^{k}$$

where the A_k is the weight of the client k obtained by AHP

Results

Accuracy, Precision, Recall, and F1 score were used as evaluation criteria. Results of original Federated Averaging were compared with the approach where AHP estimated the client weights. These criteria are calculated as follows:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

$$Recall = \frac{TP}{TP + FP}$$

$$Precision = \frac{TP}{TP + FN}$$

$$F1\ Score = \frac{2 * Recall * Precision}{Recall + Precision}$$

F1-score was taken as the primary evaluation criterion since the training set for some clients is more imbalanced. After each iteration, weights were sent back to clients to evaluate their test dataset.

Experiments

For experiments, the MAGIC Gamma Telescope Dataset was used. This dataset consists of 19020 rows and 11 columns with two classes: gamma and hydron. For experimental purposes, the dataset is distributed among clients with different sizes and imbalances. The summary of clients is demonstrated in Table 3.

Table 3. Client information

Clients	Size	Class 'g'	Class 'h'	Gini	Comp. Power (GHz)
1	4000	2000	2000	0.5	4.5
2	6300	4500	1800	0.4	3.0
3	3500	2000	1500	0.48	1.5
4	1100	500	600	0.49	4.5
5	4120	3332	788	0.31	3.0

AHP returned the following weights for each client:

- Client 1 – 0.242
- Client 2 – 0.199
- Client 3 – 0.171
- Client 4 – 0.222
- Client 5 – 0.164

The following table demonstrates the accuracy, recall, precision, and f1 score of both the ahp approach and the original FedAVG:

Table 4 Results and comparisons. Numbers in bold show the better result between the original and our approach

Device	Original				Our approach			
	Acc	Rec	Pre	F1	Acc	Rec	Pre	F1
1	0.70	0.77	0.54	0.64	0.73	0.75	0.61	0.67
2	0.76	0.60	0.52	0.56	0.76	0.57	0.60	0.59
3	0.68	0.64	0.47	0.55	0.71	0.66	0.59	0.62
4	0.68	0.83	0.53	0.65	0.69	0.80	0.58	0.67
5	0.81	0.47	0.57	0.51	0.79	0.43	0.66	0.52
Aver-age	0.72	0.66	0.53	0.58	0.73	0.64	0.61	0.61

From the results, recall score is generally better in original weighting. However, except for one client, the AHP approach outperforms in Precision, F1 score, and accuracy.

F1 curves are demonstrated in the following figure. For each iteration, we can see that the AHP approach demonstrates better results and converges faster. It can be caused by taking computation power into account.

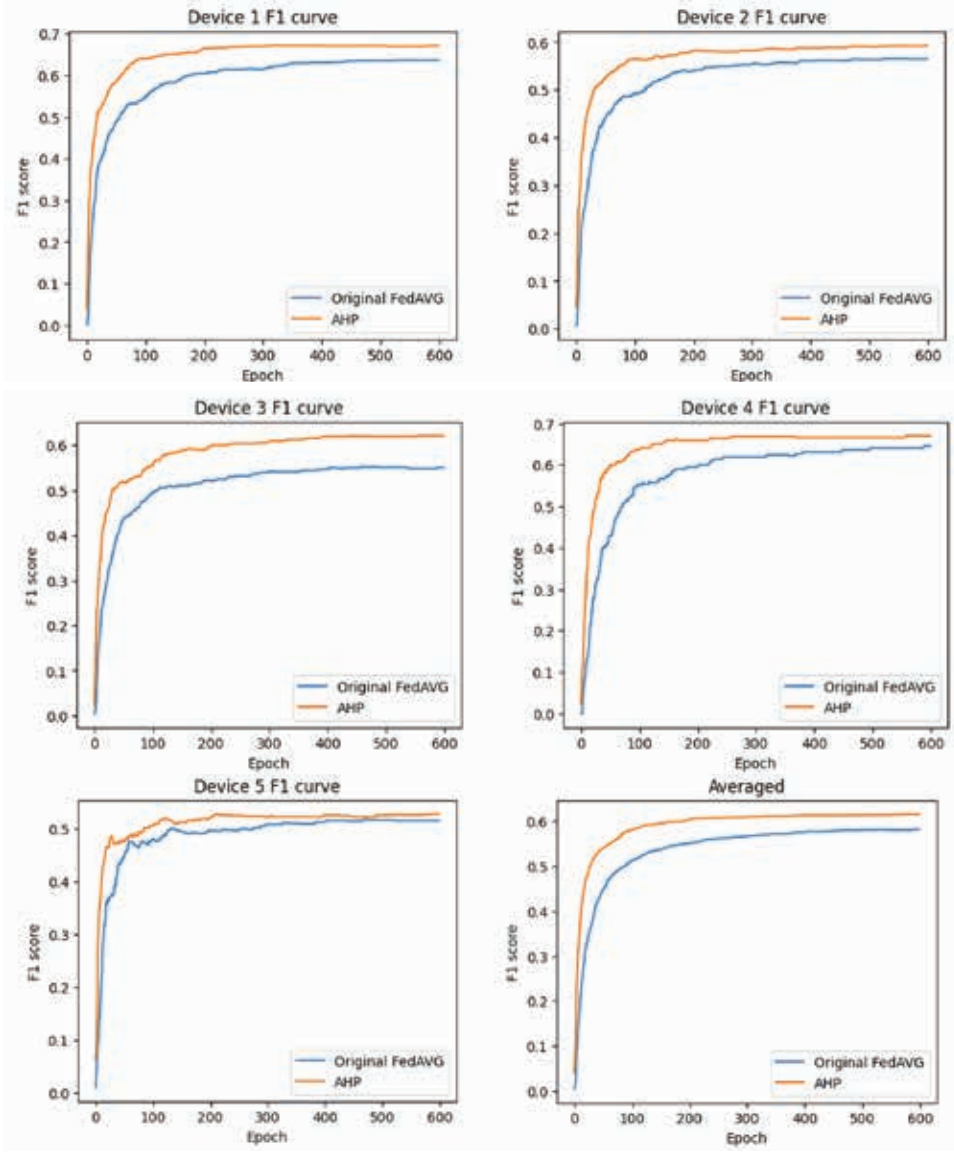


Fig. 5: F1 curves. (a),(b),(c),(d),(e) shows the curves of clients respectively. The curve in the (f) represents the average F1 score of clients.

Conclusion and Future Works

Results show that adding other criteria to deciding the weights performed better than the FedAVG algorithm, which only considers the dataset size. Using AHP also made the global model converge faster. The Only downside was the recall score decline, whereas other metric measures were better.

Work dedicated to increasing the recall score can be done in the future. Other open research questions are how it will perform if the number of clients is bigger. The preference matrix discussed in the Methodology section is open to be modified. It can be done by changing preferences or adding other criteria. Applying this to bigger models, such as neural networks, is another work that can be an extension of this research.

References

Aliyev, S., & Ismayilova, N. (2023, October). FL2: Fuzzy Logic for Device Selection in Federated Learning. In *2023 IEEE 17th International Conference on Application of Information and Communication Technologies (AICT)* (pp. 1-6). IEEE.

Du, Z., Wu, C., Yoshinage, T., Zhong, L., & Ji, Y. (2022, October). On-device federated learning with fuzzy logic based client selection. In *Proceedings of the Conference on Research in Adaptive and Convergent Systems* (pp. 64-70).

Hong, M., Kang, S. K., & Lee, J. H. (2022). Weighted averaging federated learning based on example forgetting events in label imbalanced non-iid. *Applied Sciences, 12(*12), 5806.

Li, Y., Guo, Y., Alazab, M., Chen, S., Shen, C., & Yu, K. (2022). Joint optimal quantization and aggregation of federated learning scheme in VANETs. *IEEE Transactions on Intelligent Transportation Systems, 23*(10), 19852-19863.

Qi, P., Chiaro, D., Guzzo, A., Ianni, M., Fortino, G., & Piccialli, F. (2023). Model aggregation techniques in federated learning: A comprehensive survey. *Future Generation Computer Systems.*

Roberts, G., Rao, N. K., & Kumar, S. (1987). Logistic regression analysis of sample survey data. *Biometrika, 74*(1), 1-12.

Saaty, R. W. (1987). The analytic hierarchy process—what it is and how it is used. *Mathematical modelling, 9*(3-5), 161-176.

Sun, T., Li, D., & Wang, B. (2022). Decentralized federated averaging. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 45*(4), 4289-4301.

Tang, Z., Shao, F., Chen, L., Ye, Y., Wu, C., & Xiao, J. (2021). Optimizing federated learning on non-IID data using local Shapley value. In *Artificial Intelligence: First CAAI International Conference, CICAI 2021, Hangzhou, China, June 5–6, 2021, Proceedings, Part II 1* (pp. 164-175). Springer International Publishing.

Wilbik, A., & Grefen, P. (2021, July). Towards a federated fuzzy learning system. In *2021 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)* (pp. 1-6). IEEE.

Xu, C., Hong, Z., Huang, M., & Jiang, T. (2022). Acceleration of federated learning

with alleviated forgetting in local training. *arXiv preprint arXiv:2203.02645.*

Yadav, A., & Jayswal, S. C. (2013). Using geometric mean method of analytical hierarchy process for decision making in functional layout. *International Journal of Engineering Research and Technology (IJERT), 2*(5).

Ye, R., Xu, M., Wang, J., Xu, C., Chen, S., & Wang, Y. (2023). FedDisco: Federated Learning with Discrepancy-Aware Collaboration. *arXiv preprint arXiv:2305.19229.*

Submitted 19.09.2023
Accepted 07.11.2023

Azerbaijan Journal of High Performance Computing, Vol 6, Issue 2, 2023, pp. 163-170
https://doi.org/10.32010/26166127.2023.6.2.163.170

Survey of Usage Artificial Intelligence Mechanism in the Load Balancer

Nigar Ismayilova

Azerbaijan State Oil and Industry University, Baku, Azerbaijan, nigar.ismailova@asoiu.edu.az

*Correspondence: Nigar Ismayilova, Azerbaijan State Oil and Industry University, Baku, Azerbaijan, nigar.ismailova@asoiu.edu.az

Abstract

Nowadays, there is no way to imagine artificial intelligence applications without using high-performance computing systems. The huge amount of processing data, the complex structure of learning technology, time limitations, and the necessity of real-time operation require powerful computational resources and parallel algorithms. This paper analyzed another direction of convergence between high-performance computing and artificial intelligence: using artificial intelligence techniques in one of the main problems of distributed systems load balancing. The primary objective of this work is to examine the necessity of using AI concepts in load balancing and the definition of providing facilities for load balancers.

Keyword: Load Balancer, Convergence of HPC and AI, Dynamic Load Balancer, Task Scheduling, Artificial Intelligence.

1. Introduction

By increasing the usage of HPC technologies in various fields of science and industry optimization of load balancers, finding the best assignment between processes and resources has become challenging. The challenges with optimization methods for task scheduling in different types of distributed computing systems are growing with the progress of computational resources and scientific inventions. The optimal distribution of tasks between resources in cluster computing systems with the standing quantity of requests and computational machines based on general network optimization methods such as bipartite matching, minimax criteria, and finite element methods have been successfully applied (Chu, W. C., Yang, D. L., Yu, J. C., & Chung, Y. C., 2001; Shen, C. C., & Tsai, W. H., 1985; Harvey, N. J., Ladner, R. E., Lovász, L., & Tamir, T., 2006). On the other hand, new approaches based on artificial intelligence must be applied to dynamic environments with unstable resources and requests. The goal of a load balancer in cloud computing systems is the optimization of comprehensive computing capacity, which differs from the load balancer's maximization of computing mission in other distributed systems such as cluster computing, grid computing, peer-to-peer computing, and exascale computing (Ramya & Senthilselvi, 2021). With the popularization of modern wireless communications, load balancers in fog computing systems are necessary. The goal of load balancers in such systems is to minimize execution time and energy consumption.

This work investigates different approaches for optimizing load balancers based on

artificial intelligence techniques, indicates the advantages and disadvantages of proposed methods and their applications, and points out challenges for getting better results in optimizing load balancers.

Section 2 analyses the classification of load balancers in computing systems using different parameters, the third section is concerned with different approaches for minimization of execution time and cost based on machine learning algorithms, and the fourth section presents the application of fuzzy logic-based methods for task scheduling, pros and cons of these approaches. Section five has demonstrated using genetic algorithms to find the best solutions in task assignments for distributed computing systems. The paper ends with a discussion and conclusion sections.

2. Classification of Load Balancers

The mechanism of load balancing activity can be modeled by the following formula (Bakhishoff, U., Khaneghah, E. M., Aliev, A. R., & Showkatabadi, A. R., 2020):

Accordingly, this formula describes conditions for optimal load balancing. It is an assignment process of processes to computing machines where each process in the system must be scheduled, and the activity of all resources should be 100 percent.

There are different approaches and parameters for the classification of load balancer strategies. Some 3researchers, as essential parameters, use the location for load balancers and classify them as centralized and distributed (Waraich, 2008) or centralized, decentralized, and hierarchical load balancers (Al-Rayis & Kurdi, 2013). Several methods and algorithms have been proposed for implementing these strategies in multi-scale computing systems, and their priorities were justified by experiments (Barazandeh & Mortazavi). There is also a large body of work considering the classification of load balancers into local and global classes based on the information needed for distributing requests. The superiority of distributed algorithms for load balancers, which gets information from the global state, has been demonstrated over local load balancers, showing unscalable distribution in the computing system (Gasmelseed & Ramar, 2019).

A more general classification of load balancers as static and dynamic has been proposed based on system characteristics. For static load balancers where several available resources and solving requests are persistent were proposed application of famous scheduling algorithms such as Round Robin, least connections approach, and graph matching (Alankar, B., Sharma, G., Kaur, H., Valverde, R., & Chang, V., 2020; Wei, L. F., Ji, J. W., & Zhao, L. Q., 2011; Kaur, M., & Mohana, R., 2019; Devi, R. K., & Muthukannan, M., 2018, October). Some studies proposed a partitioning approach to distributing unmanageable loads between resources in static systems (Meyerhenke, H., Monien, B., & Sauerwald, T., 2009; Sevilla, M. A., et al., 2015, November). All these approaches have their advantages and limitations. The main disadvantage of the round-robin approach is an assumption about equality of resource abilities and the nonexistence of perspectives for AI applications (Aza & Urrea, 2019). Randomized load balancing established on the base of the Poisson process or Markov process application of statistical inference algorithms

and state-based search AI methods will provide efficient results for load balancing optimization (Bramson et al., 2010). Centralized load balancers can improve their activity by applying different clustering methods for classification and learning the processes in the computing system. The machine learning approach allows us to optimize the load balancing in dynamic, decentralized Grid systems by threshold approach (Rathore, N., 2016; Goldsztajn, D., et al., 2022; Lin, W., Wang, J. Z., Liang, C., & Qi, D., 2011; Rathore, N., & Chana, I., 2015). The main problem here, as usual in systems with dynamic structures, is the recognition and classification of the unlabeled data. AI forecasting methods open wide areas for high-performance load-balancing models based on least-connection scheduling algorithms (Choi, D., Chung, K. S., & Shon, J., 2010, December; Ren, X., Lin, R., & Zou, H., 2011, September). State-based techniques of AI, especially statistical inference methods, can effectively be applied to the organization of load balancing in distributed systems by local queue algorithms and central queue algorithms (Sharma, S., Singh, S., & Sharma, M., 2008).

Progress and intensive usage of heterogeneous computing environments such as grid computing, P2P computing, and cloud computing, as well as challenges in Exascale computing systems, demonstrate the necessity and effectiveness of proposed dynamic load balancers for the distribution of load (Chandakanna, V. R., & Vatsavayi, V. K., 2016; Rajavel, R., Somasundaram, T. S., & Govindarajan, K., 2010). Several investigations used artificial intelligence techniques to assign requests to suitable resources where one or both have a dynamic nature (Hongvanthong, S., 2020, May; Nadaph, A., & Maral, V., 2015, February). The following sections have discussed different approaches for optimizing load balancers in computing systems with a dynamic nature based on artificial intelligence methods.

3. Machine Learning Based Methods for Load Balancer

The use of machine learning algorithms for the selection of the most effective load-balancing algorithm in heterogeneous HPC systems with requests demonstrating dynamic and unknown behavior has been discussed by a significant number of authors (Oikawa, C. A. V., Freitas, V., Castro, M., & Pilla, L. L., 2020, March). Researchers attempted to reduce execution time and communication load and guarantee real-time scheduling by application of different types of neural networks, as well as reinforcement learning. Generally, in these methods, optimal load distribution by the load balancer and real-time decision-making are implemented by learning based on information received from a dynamic computing environment and feedback information. Ahmed et al. use classification methods based on different features extracted from the characteristics of scientific applications for load balancing in heterogeneous computing systems. For training and testing a database comprising execution results of parallel applications. This approach reduces execution time, increases the resource utilization ratio, and performs better than other scheduling algorithms. The main limitation of these methods is the impossibility of increasing the database to the desired volume to improve the accuracy of the classification system.

The application of clustering algorithms for load balancer optimization is widely reported in the literature by researchers. The authors used a k-means clustering algorithm to reduce resource costs and execution time. Conceptually similar work implemented by Sun and others, which used improved k-means clustering according to resource attributes information, which helps to reduce the search environment during task scheduling (Sun et al., 2014). Despite the success of the mentioned approach for load balancer optimization in distributed computing environments, it is still limited by the narrow nature of resource attributes. Using fuzzy c-means clustering and applying this method to classify resources and tasks can be considered for more 'soft' task scheduling in heterogeneous computing systems.

Mao and others reported on a new approach based on the Bayesian model for optimization of resource usage in cloud environments. Their work traces the advantages of predicting prior probabilities using posterior information for optimization load balancer's work (Mao et al., 2014). Bayesian model allows the management work of load balancers in heterogeneous systems. On the other hand, the optimization process is realized by considering a few characteristics of the resources and processes. There is a necessity for a detailed analysis of process and resource characteristics for their appropriate classification.

4. Fuzzy Logic-based Approaches for Load Balancer

Fuzzy inference system (FIS) and adaptive neuro-fuzzy inference system are the main methods based on fuzzy sets theory for optimizing the load distribution in dynamic and heterogeneous environments. Computing with fuzzy logic-based methods and the possibility of handling uncertainties using these algorithms allows us to make real-time decisions in diverse computing systems. Using logic-based artificial intelligence techniques ensures clear information about optimizing the load-balancing process, which can easily be improved through regular monitoring. For, Setia and others applied fuzzy logic-based load balancing for parallel master-slave implementations with linguistic variables for input (estimated load and estimated delay in the nodes) and output (takeover capacity of the system) (Setia et al., 2009).

As has been investigated in some works, the application of logic-based techniques of artificial intelligence, such as using fuzzy logic controllers for fault tolerance management in cloud systems, managing traffic in fog computing systems, and stabilizing load among computational resources in multiprocessor systems, gives appropriate solutions, this also has been used ANFIS load balancer for optimization in Big Data. This approach can demonstrate better performance and require less effort if the *parameters of the system can be determined from the learning process. However, it still needs a considerable amount of data about executed applications.

5. Algorithms Based on Evolutionary Algorithms Used for Load Balancer Optimization

One of the most well-known approaches for handling decentralized computing systems

is using different naturally inspired algorithms based on swarm intelligence. Dasgupta and others examined the optimization of load balancers in cloud computing using genetic algorithms. This approach has been selected as the best scheduling by minimizing the execution time of requests (Dasgupta et al., 2013). Different approaches based on genetic algorithms are used for multicriteria scheduling tasks, regularization of traffic in the network, minimization of energy consumption, and another objective of the load balancer.

G. Sivashanmugam and others focused on the problems of load balancer development in cloud computing systems; they have demonstrated that besides the primary goal of the load balancer to assign tasks to resources, we also need problem-solving and computing abilities. For this purpose, authors have proposed a naturally inspired load balancer algorithm entirely based on eagle behavior (Sivashanmugam et al., 2019). Although approaches based on evolutionary algorithms for task scheduling in computing environments demonstrate success in achieving the primary goal, search in the big environment cannot propose an optimal solution. For this purpose, combining logic-based approaches with evolutionary algorithms can give better results.

6. Discussion

As mentioned in the central part of the work, there are many opportunities for improving the task scheduling process in distributed computing systems by using different methods and algorithms of artificial intelligence. However, the application results of these approaches could be more satisfactory. They have various disadvantages that can be eliminated by using different combinations of the mentioned approaches or proposing novel ideas for handling load balancing in computing systems with a dynamic nature drawing on logic-based artificial intelligence techniques.

Conclusion

This study aimed to evaluate different approaches for load balancers in distributed computing systems, especially systems with dynamic resources and requests, using artificial intelligence mechanisms. The methods reported here, with their priorities and shortcomings, give new objectives for improving the task scheduling process by proposing innovative conceptions. More research for real-time decision-making in heterogeneous computing systems using intelligent approaches is needed to optimize the job scheduling process. This study suggests using intelligent methods for minimization of execution time and computing cost in multi-scale computing systems, which will be able to demonstrate good learning performance using a smaller amount of historical information and environmental feedback. For this reason, applying Bayesian inference methods can demonstrate high accuracy by recovering missing data. On the other hand, using soft clustering methods can reduce search space for tackling the load balancer.

References

Al-Rayis, E., & Kurdi, H. (2013). Performance Analysis of Load Balancing Archi-

tectures in Cloud Computing [Proceedings Paper]. Uksim-Amss Seventh European Modelling Symposium on Computer Modelling and Simulation (Ems 2013), 520-524. https://doi.org/10.1109/ems.2013.10

Alankar, B., Sharma, G., Kaur, H., Valverde, R., & Chang, V. (2020). Experimental setup for investigating the efficient load balancing algorithms on virtual cloud. Sensors, 20(24), 7342.

Aza, E. F., & Urrea, J. P. (2019). Implementation of Round-Robin load balancing scheme in a wireless software defined network [Proceedings Paper]. 2019 Ieee Colombian Conference on Communications and Computing (Colcom 2019), 6.

Bakhishoff, U., Khaneghah, E. M., Aliev, A. R., & Showkatabadi, A. R. (2020). DTHMM ExaLB: discrete-time hidden Markov model for load balancing in distributed exascale computing environment. Cogent Engineering, 7(1), 1743404.

Barazandeh, I., & Mortazavi, S. S. (2009, Dec 28-30). Two Hierarchical Dynamic Load Balancing Algorithms in Distributed Systems.International Conference on Computer and Electrical Engineering ICCEE [Second international conference on computer and electrical engineering, vol 1, proceedings]. 2nd International Conference on Computer and Electrical Engineering, Dubai, U ARAB EMIRATES.

Bramson, M., Lu, Y., Prabhakar, B., & Acm. (2010). Randomized Load Balancing with General Service Time Distributions [Proceedings Paper]. Sigmetrics 2010: Proceedings of the 2010 Acm Sigmetrics International Conference on Measurement and Modeling of Computer Systems, 38(1), 275-286.

Chandakanna, V. R., & Vatsavayi, V. K. (2016). A QoS-aware self-correcting observation based load balancer. Journal of Systems and Software, 115, 111-129.

Choi, D., Chung, K. S., & Shon, J. (2010, December). An improvement on the weighted least-connection scheduling algorithm for load balancing in web cluster systems. In International Conference on Grid and Distributed Computing (pp. 127-134). Berlin, Heidelberg: Springer Berlin Heidelberg.

Chu, W. C., Yang, D. L., Yu, J. C., & Chung, Y. C. (2001). UMPAL: an unstructured mesh partitioner and load balancer on World Wide Web. J. Inf. Sci. Eng., 17(4), 595-614.

Dasgupta, K., Mandal, B., Dutta, P., Mondal, J. K., & Dam, S. (2013). A Genetic Algorithm (GA) based Load Balancing Strategy for Cloud Computing [Proceedings Paper]. First International Conference on Computational Intelligence: Modeling Techniques and Applications (Cimta) 2013, 10, 340-347. https://doi.org/10.1016/j.protcy.2013.12.369

Devi, R. K., & Muthukannan, M. (2018, October). Mobile Agent-based Secure Cloud Data Center Exploration for Load Data Retrieval Using Graph Theory. In Proceedings of the 2018 International Conference on Cloud Computing and Internet of Things (pp. 1-6).

Gasmelseed, H., & Ramar, R. (2019). Traffic pattern-based load-balancing algorithm in software-defined network using distributed controllers [Article]. International

Journal of Communication Systems, 32(17), 14, Article e3841. https://doi.org/10.1002/dac.3841

Goldsztajn, D., et al. (2022). Self-learning threshold-based load balancing. INFORMS Journal on Computing, 34(1), 39-54.

Harvey, N. J., Ladner, R. E., Lovász, L., & Tamir, T. (2006). Semi-matchings for bipartite graphs and load balancing. Journal of Algorithms, 59(1), 53-78.

Hongvanthong, S. (2020, May). Novel four-layered software defined 5g architecture for ai-based load balancing and qos provisioning. In 2020 5th International Conference on Computer and Communication Systems (ICCCS) (pp. 859-863). IEEE.

Kaur, M., & Mohana, R. (2019). Static load balancing technique for geographically partitioned public cloud. Scalable Computing: Practice and Experience, 20(2), 299-316.

Lin, W., Wang, J. Z., Liang, C., & Qi, D. (2011). A threshold-based dynamic resource allocation scheme for cloud computing. Procedia Engineering, 23, 695-703.

Mao, H. Y., Yuan, L., & Qi, Z. W. (2014). A Load Balancing and Overload Controlling Architecture in Clouding Computing [Proceedings Paper]. 2014 Ieee 17th International Conference on Computational Science and Engineering (Cse), 1589-1594. https://doi.org/10.1109/cse.2014.293

Meyerhenke, H., Monien, B., & Sauerwald, T. (2009). A new diffusion-based multilevel algorithm for computing graph partitions. Journal of Parallel and Distributed Computing, 69(9), 750-761.

Nadaph, A., & Maral, V. (2015, February). Methodical analysis of various balancer conditions on public cloud division. In 2015 International Conference on Computing Communication Control and Automation (pp. 40-46). IEEE.

Oikawa, C. A. V., Freitas, V., Castro, M., & Pilla, L. L. (2020, March). Adaptive load balancing based on machine learning for iterative parallel applications. In 2020 28th Euromicro International Conference on Parallel, Distributed and Network-Based Processing (PDP) (pp. 94-101). IEEE.

Rajavel, R., Somasundaram, T. S., & Govindarajan, K. (2010). Dynamic load balancer algorithm for the computational grid environment. In Information and Communication Technologies: International Conference, ICT 2010, Kochi, Kerala, India, September 7-9, 2010. Proceedings (pp. 223-227). Springer Berlin Heidelberg.

Ramya, K., & Senthilselvi, A. (2021). Performance Improvement in Cloud Computing Environment by Load Balancing-A Comprehensive Review. Revista Geintec-Gestao Inovacao E Tecnologias, 11(2), 1386-1399.

Rathore, N. (2016). Dynamic threshold based load balancing algorithms. Wireless Personal Communications, 91(1), 151-185.

Rathore, N., & Chana, I. (2015). Variable threshold-based hierarchical load balancing technique in Grid. Engineering with computers, 31, 597-615.

Ren, X., Lin, R., & Zou, H. (2011, September). A dynamic load balancing strategy for cloud computing platform based on exponential smoothing forecast. In 2011 IEEE

international conference on cloud computing and intelligence systems (pp. 220-224). IEEE.

Setia, A., Swarup, V. M., Kumar, S., Singh, L., & Ieee. (2009). A Novel Adaptive Fuzzy Load Balancer for Heterogeneous LAM/MPI Clusters Applied to Evolutionary Learning in Neuro-Fuzzy Systems [Proceedings Paper]. 2009 Ieee International Conference on Fuzzy Systems, Vols 1-3, 68-+. https://doi.org/10.1109/fuzzy.2009.5277322

Sevilla, M. A., et al. (2015, November). Mantle: a programmable metadata load balancer for the ceph file system. In Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis (pp. 1-12).

Sharma, S., Singh, S., & Sharma, M. (2008). Performance analysis of load balancing algorithms. International Journal of Civil and Environmental Engineering, 2(2), 367-370.

Shen, C. C., & Tsai, W. H. (1985). A graph matching approach to optimal task assignment in distributed computing systems using a minimax criterion. IEEE Transactions on Computers, 100(3), 197-203.

Sivashanmugam, G., Shantharajah, S. P., & Iyengar, N. (2019). Avian Based Intelligent Algorithm to Provide Zero Tolerance Load Balancer for Cloud Based Computing Platforms [Article]. International Journal of Grid and High Performance Computing, 11(4), 42-67. https://doi.org/10.4018/ijghpc.2019100104

Sun, X. Y., Fu, X. L., Hu, H., & Gui, T. (2014). The Cloud computing tasks scheduling algorithm based on improved K-Means. Applied Science, Materials Science and Information Technologies in Industry, 513-517, 1830-1834. https://doi.org/10.4028/www.scientific.net/AMM.513-517.1830

Waraich, S. S. (2008). Classification of dynamic load balancing strategies in a network of workstations [Proceedings Paper]. Proceedings of the Fifth International Conference on Information Technology: New Generations, 1263-1265.

Wei, L. F., Ji, J. W., & Zhao, L. Q. (2011). The Research and Design of Two-Level Load Balancer Based on Web Server Cluster. Advanced Materials Research, 282, 765-769.

Submitted 21.09.2023
Accepted 15.11.2023

Azerbaijan Journal of High Performance Computing, Vol 6, Issue 2, 2023, pp. 171-190
https://doi.org/10.32010/26166127.2023.6.2.171.190

Thai Text-to-Image Prompt Engineering by Pre-trained Large Language with Stable Diffusion Model

Pakpoom Mookdarsanit, Lawankorn Mookdarsanit

Chandrakasem Rajabhat University, Bangkok, Thailand, pakpoom.m@chandra.ac.th, lawankorn.s@chandra.ac.th

*Correspondence: Pakpoom Mookdarsanit, Chandrakasem Rajabhat University, Bangkok, Thailand, pakpoom.m@chandra.ac.th,

Abstract

Text-to-image (T2I) generation is a new area of large language models (LLMs), a type of prompt engineering involving inputting a textual description to generate an image. To shift a new paradigm of Thai natural language processing (Thai-NLP), this paper first presents state-of-the-art Thai Text-to-Image prompt engineering (TH-T2I) to translate Thai text into a semantic image according to the semantic Thai textual description. The pre-trained SCB-MT-EN-TH model is employed for Text-to-Text (T2T) translation. Moreover, the image generation is done according to a semantic text prompt by a stable diffusion model. The T2T is evaluated by Bi-lingual Evaluation Understudy (BLEU), while T2I is done by Inception and Frechet Inception Distance (FID). The images generated by TH-T2I were of high quality, as measured by Inception and FID. TH-T2I contributes to a T2I baseline model in Thai, preserving the Thai cultural language on digital heritage.

Keyword: Text-to-Image Translation, Thai Prompt Engineering, Stable Diffusion Model, Image Generation.

1. Introduction

Although many large language models (LLMs) have been recently introduced, some low-resource languages must be generative artificial intelligence (AGI). Unlike English (one of the high-resource languages), Thai is one of the low-resource languages (Arreerard, Mander & Piao, 2022). that is locally spoken in Thailand and the surrounding Mekong Golden Triangle region. As to some LLMs and Thai, there were three well-known Transformer-based models: WangchanBERTa - a pre-trained Thai text categorization (Lowphansirikul, Polpanumas, Jantrakulchai & Nutanong, 2021)., as well as PhayaThaiBERT (Sriwirote, Thapiang, Timtong & Rutherford, 2023). and SCB-MT-EN-TH model (Lowphansirikul, Polpanumas, Rutherford & Nutanong, 2020). - a large-scale Thai-English machine translation (Lowphansirikul, Polpanumas, Rutherford & Nutanong, 2022). To enhance the conventional T2T translation model (e.g., text categorization, TH-EN machine translation) into T2I, Thai Text-to-Image (TH-T2I), prompt engineering was presented as one of the new challenges of Thai natural language processing areas that

are proposed to translate the Thai text into an image. For the academic movement beyond education (Mookdarsanit & Mookdarsanit, 2022)., TH-T2I could be a baseline model in T2I on Thai for future Thai AI competitions (e.g., BEST Hackathon organized by NECTEC) that were proposed to preserve Thai as the digital heritage inherited by the next generation of Thai AGI researchers. As well as planting the Thai digital Treebank, future digital donations might be in Thai speech, handwriting, or textual comments on social media to make Thai a fruitful resource and library comparable to other high-resource languages.

1.1. Thai-NLP Timeline

Thai (one of the Kra-dai family) has been used as a local language in Thailand and the Mekong Golden Triangle region for longer than 720 years since the Sukothai stone tablet inscribed in Thai scripts by King Ramkhamhaeng (Inthajakra, Prachyapruit & Chantavanich, 2016). The tablet has been declared a World Heritage by UNESCO since 2003. Thai vocabulary acquired from Sanskrit, Pali, Khmer, and Mon. Linguistically, Thai is a native language spoken by 70 million people. Thai is a tonal language in speech that implies different meanings, which is one of the challenges in the 5G penetration test for Thai tonal speech. To give an example, the word “Ma” in Thai has five different tones: the first tone (Thai: มา, v.) means coming, the second tone (Thai: ม่า, n.) means grandmother, the third tone (Thai: ม้า, n.) as horse, the fourth (Thai: ม๊า, n.) as mother and fifth tone (Thai: หมา, n.) as dog, respectively. In addition, all scripts within a Thai text have no space between 2 words or phrases (Lapjaturapit, Viriyayudhakom & Theeramunkong, 2018). that needs some tokenization algorithm for separating between words/phrases (Klahan, Pannoi, Uewichitrapochana & Wiangsripanawan, 2018).

The first idea of Thai-English (TH-EN) machine translation (Koanantakool, Karoonboonyanan & Wutiwiwatchai, 2009). The Multilingual Machine Translation project (MMT) was proposed to automatically interpret between 2 languages in 1987 by Thailand's National Electronics and Computer Technology (NECTEC) as the origin of Thai natural language processing (Thai-NLP). The first LEXiTRON by NECTEC was launched in 1995 and had 11,000 Thai and 9,000 English entries (Sornlertlamvanich, 2019). Simultaneously, not only was the first Thai font introduced, known as Thai optical character recognition (Thai-OCR), but also the more complex Thai handwritten digit recognition (Thai-HDR). Both Thai-OCR (Emsawas & Kijsirikul, 2016). and Thai-HDR (Mookdarsanit & Mookdarsanit, 2020b). It could be seen as the Thai NLP meets Computer Vision. In the deep learning age, Thai-OCR had no more challenges (recognizing Thai characters from the image and understanding semantic text). Thai-OCR was expanded to meme image categorization (Mookdarsanit & Mookdarsanit, 2021a). Or scene text detection (Kobchaisawat, Chalidabhongse & Satoh, 2020). Thai-HDR recognizes the variety of Thai handwriting styles and generates different Thai handwriting styles written by AGI (Mookdarsanit & Mookdarsanit, 2021b).

From the previous literature, Thai-NLP has lasted longer than 35 years (Sornlertlamvanich, 2019). Researchers from NECTEC, Thammasat University, and Chulalongkorn University published many state-of-the-art Thai-NLP papers, respectively. Since the NECTEC was

founded with the vision of human language and computer (to make computers understand the Thai language in terms of "text," "speech", or "image"), many projects were provided for Thai industry (Tapsai, Unger & Meesad, 2020)., e.g., AI for Thai, OAM (a framework for design an ontology), Blackbeard Treebank, VAJA (as Text-to-Speech translation), BEST Hackathon. In 2000, a Thai character cluster (TCC) was designed to digitally group Thai characters (Theeramunkong, Sornlertlamvanich, Tanhermhong & Chinnan, 2000) by researchers from Thammasat University which highly impacted new Thai font construction (e.g., Chulabhorn Likit Font - dedicated to Princess Chulabhorn's continuously work for cancer patients in Thailand). Many researchers from Chulalongkorn University also played the leading role in contributing to Thai-NLP publications and available projects: AKARAWISUT (as a Thai plagiarism detector), GOWAJEE (Thai Speech Recognition), Thai-dependency Treebank, and Thai Speech-emotion Dataset.

Above all, Thai-NLP areas have been continuously developed. They could be categorized into text categorization (Mookdarsanit & Mookdarsanit, 2019)., sentiment analysis (Haruechaiyasak, Kongthon, Palingoon & Trakultaweekoon, 2013)., part of speech tagging (Boonkwan & Supnithi, 2017)., human resource language intelligence (Mookdarsanit & Mookdarsanit, 2020b)., plagiarism detection (Taerungruang & Aroonmanakun, 2018)., news summarization (Ketui, Theeramunkong & Onsuwan, 2013)., fake news detection (Mookdarsanit & Mookdarsanit, 2021b)., and TH-EN machine translation (Lowphansirikul, Polpanumas, Rutherford & Nutanong, 2022). To shift a new paradigm of Thai-NLP area, this paper presented a novel Thai Text-to-Image (TH-T2I) prompt engineering to generate an image according to Thai text as contributing a new baseline LLM enhancing from TH-EN machine translation.

1.2. Presented TH-T2I as a New Paradigm of Thai-NLP

TH-EN machine translation could be seen as a text-to-text (T2T) translation that took a long time for the model to be completed entirely. Fluency and adequacy were the T2T translation measurements. From 2006 to 2016, there were attempting to design the completed TH-EN machine translation in the domain of law (Tirasaroj, 2016). and stock exchange (Ruangrajitpakorn, 2006). from Chulalongkorn University. However, statistical machine translation (SMT) was not enough for a significant linguistic data era, and TH-EN machine translation required deep neural network models, known as large language models (LLMs). Researchers from the Vidyasirimedhi Institute of Science and Technology (VISTEC) played the leading role in LLMs for Thai-NLP. In 2020, VISTEC researchers introduced a Transformer-based translator called the "SCB-MT-EN-TH model" (Lowphansirikul, Polpanumas, Rutherford & Nutanong, 2022). that were pre-trained by 1,001,752 TH-EN parallel texts. Although either ChatGPT or Google Bard were pre-trained by larger-scale text, the SCB-MT-EN-TH model was constructed by native researchers (Lowphansirikul, Polpanumas, Rutherford & Nutanong, 2020).

Some TH-EN large-language translation machine models were available as Text-to-Text (T2T) prompt engineering. To shift a new paradigm of Thai-NLP, this paper expanded

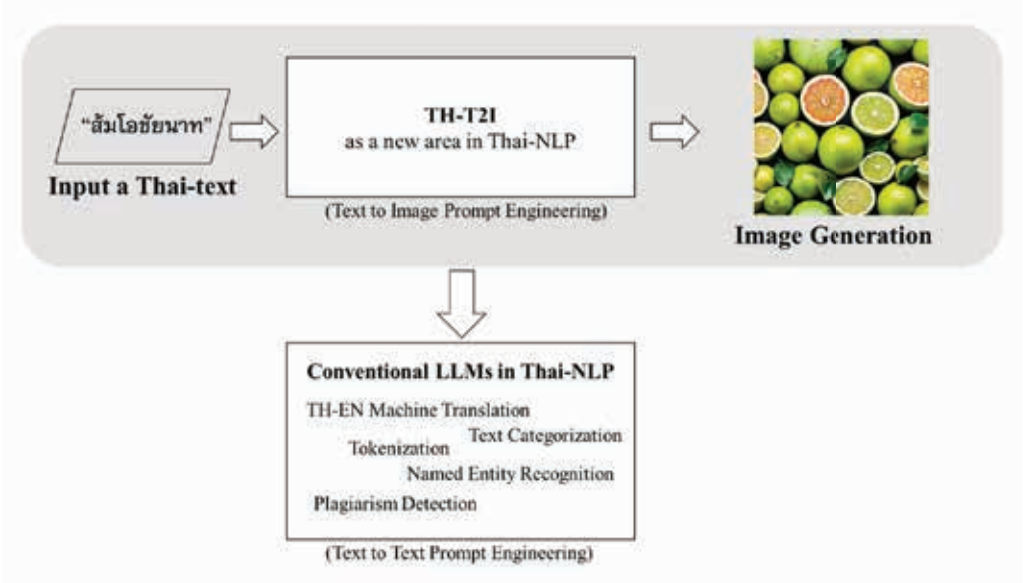


Fig. 1: They presented TH-T2I as a new paradigm of Thai-NLP research.

T2T to Text-to-Image (T2I) translation. T2I was an inverted image captioning (IC). Thai image captioning (Thai-IC) was interesting (Mookdarsanit & Mookdarsanit, 2020a) before the birth of diffusion models (Ho, Jain & Abbeel, 2020). IC was to generate a textual caption from an input image, while T2I generated an image from a textual description. The image generation was done by a stable diffusion model (Rombach, Blattmann, Lorenz, Esser & Ommer, 2021). The presented algorithm was called Thai Text-to-Image prompt engineering (TH-T2I), which was the meeting between Thai-NLP and computer vision (Lee, Hoover, Strobelt, Wang, Peng, Wright, Li, Park, Yang, Chau, 2023). As motivated by Thai-dessert image synthesis (Mookdarsanit & Mookdarsanit, 2018d)., TH-T2I could be further enhanced in many semantic Image-Text relations in Thai culture and art (Mookdarsanit & Rattanasiriwongwut, 2017c). Available on large-scale social media: street surveillance (Sutthaluang & Prakancharoen, 2020)., plant recognition (Mookdarsanit & Mookdarsanit, 2019b)., image location estimation (Mookdarsanit & Rattanasiriwongwut, 2017b)., Buddhism and temples (Mookdarsanit & Rattanasiriwongwut, 2017a)., food image description (Soimart & Mookdarsanit, 2017a)., tourism classification (Mookdarsanit & Mookdarsanit, 2018c)., GPS place estimation (Soimart & Mookdarsanit, 2017b), pesticide analytics (Sutthaluang, 2019)., agricultural product quality (Mookdarsanit & Mookdarsanit, 2021a)., facial verification and recognition (Soimart & Mookdarsanit, 2016)., and cosmetic recommender systems (Mookdarsanit & Mookdarsanit, 2023).

The significant contribution of TH-T2I is abridged as:

- TH-T2I shifted a new paradigm of Thai-NLP research areas that would be one of Thai-NLP timelines (as well as Thai-IC).
- TH-T2I expanded from T2T to T2I translation that combined Thai-NLP and stable

diffusion model.

- TH-T2I would be a T2I baseline model in Thai, especially in AI competitions (e.g., BEST Hackathon).

- TH-T2I supported the preservation of Thai cultural language on digital heritage (that could be inherited by the next generation of Thai AGI researchers), comparable to other high-resource languages.

This paper is organized into five parts. Attention mechanism and pre-trained significant language with a stable diffusion model were described in parts 2 and 3. Part 4 discussed experimental evaluations and results. Finally, part 5 was the conclusion.

2. Attention mechanism

Since statistical machine translation (SMT) was unsuitable for linguistic big data, Transformer architecture (Vaswani, Shazeer, Parmar, Uszkoreit, Jones, Gomez, Kaiser & Polosukhin, 2017). was introduced by Google in 2017 to handle parallel translations of a large number of languages on Google. As to the vanishing gradient in processing long text by the recurrent neural network, the Transformer was a sequence-to-sequence (seq2seq) designed for large-scale machine translation. A transformer could be seen as an encoder-decoder model, like SMT. One of the essential mechanisms in the Transformer was attention. The presented TH-T2I used self-attention (as a multi-head parallel block within the encoder and decoder) and mask self-attention (as a multi-head parallel block within the decoder) in the T2T part and self-attention in the T2I part.

2.1. Self-attention

Prior to self-attention, the attention concept was to focus the tokens (terms or words/phrases) within the text because the main idea of a sentence depended on the weight of significant tokens. By the following steps, the attention should be computed by (1)

$$Attention(q,k,v) = soft\max\left(\frac{q \bullet k^T}{\sqrt{d}}\right) \bullet v \tag{1}$$

where $Attention(\bullet)$ such a Softmax function, k as "Key matrix $\begin{bmatrix} k_0 \\ k_1 \\ k_2 \end{bmatrix}$", q as "Query matrix $\begin{bmatrix} q_0 \\ q_1 \\ q_2 \end{bmatrix}$" and v as "Value matrix $\begin{bmatrix} v_0 \\ v_1 \\ v_2 \end{bmatrix}$" ($\left(\frac{q \bullet k^T}{\sqrt{d}}\right)$ represented by $a_{i,j} \bullet v_i$)

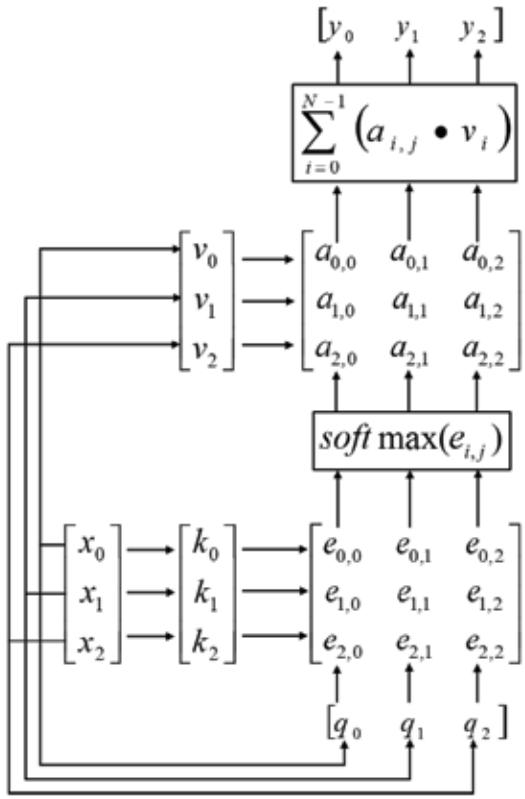


Fig. 2: Self-attention mechanism in encoder and decoder (in form of multi-head attention); and generator within TH-T2I architecture.

As to the self-attention in Figure 2, the input as "Input matrix $\begin{bmatrix} x_0 \\ x_1 \\ x_2 \end{bmatrix}$" was used to compute the "Key matrix $\begin{bmatrix} k_0 \\ k_1 \\ k_2 \end{bmatrix}$", "Query matrix $\begin{bmatrix} q_0 \\ q_1 \\ q_2 \end{bmatrix}$" and "Value matrix $\begin{bmatrix} v_0 \\ v_1 \\ v_2 \end{bmatrix}$", as (2)-(4)

$$k_i = W_k^T x_i \tag{2}$$

$$q_j = W_q^T x_i \tag{3}$$

$$v_i = W_v^T x_i \tag{4}$$

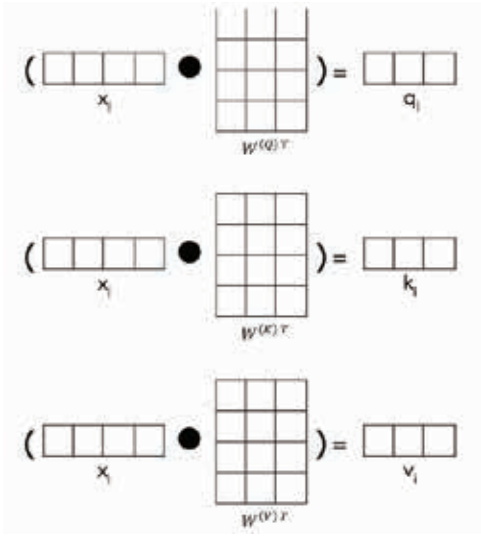


Fig. 3: The illustration of Key matrix, Query matrix and Vector matrix by (2)-(4).

First, suppose a Thai text had two tokens, token #0 and token #1; the computation explanation was in token #0. Both scores (by a2nd $q_0 k_1^T$) were assumed.

Then, in such an illustration of token #0, the score was divided by 8 (as called) since the vectors had 64 dimensions.

Next, the Softmax function would be computed to make the range value between 0 and 1.

After that, the Softmax outputs were multiplied by v_0 (in the case of token #0)

Finally, the sum of the product was computed and called. z_{00}

In TH-T2I, self-attention was a composition in the encoder and decoder side of LLM (in the form of Multi-head self-attention) and in generator stable diffusion to build an image.

2.2. Masked Self-attention

Masked self-attention was in the decoder side of LLM (in the form of Multi-head masked self-attention) in TH-T2I. The main difference between self-attention and masked self-attention was Alignment and Softmax computation.

Masked self-attention, alignment, and Softmax function computations could be defined by (5) and (6), respectively.

$$Alignment_{ij\ Masked} = \begin{cases} (Score/8)_{ij} & if\ i \geq j \\ -\infty & if\ i < j \end{cases} \quad (5)$$

$$Soft\max_{ij\ Masked} = \begin{cases} Soft\max_{ij} & if\ i \geq j \\ 0 & if\ i < j \end{cases} \quad (6)$$

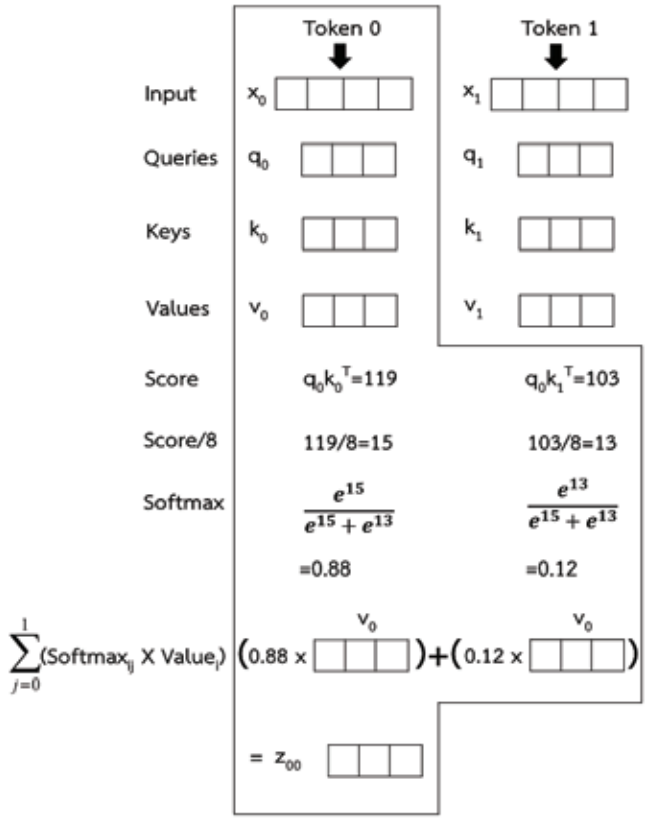


Fig. 4: The illustration of a Thai word (e.g., "ส้มโอ") represented by a token #0 attention computation.

The Masked self-attention architecture can be shown in Figure 5. Both self-attention and Masked self-attention could be in the form of a multi-head layer.

3. Pre-trained Significant Language with a Stable Diffusion Model

TH-T2I shifted a new paradigm of Thai-NLP areas by generating an image from a Thai text input; it was a combination of LLMs (Lowphansirikul, Polpanumas, Rutherford & Nutanong, 2022). and computer vision (Lee, Hoover, Strobelt, Wang, Peng, Wright, Li, Park, Yang, Chau, 2023). based on Transformer (Vaswani, Shazeer, Parmar, Uszkoreit, Jones, Gomez, Kaiser & Polosukhin, 2017). In T2T translation, the SCB-MT-EN-TH pre-trained model was used. SCB-MT-EN-TH pre-trained model was a Transformer-based model for a large-scale TH-EN machine translator that was pre-trained by 1M TH-EN texts. This pre-trained model was the best TH-EN machine translator (compared to others, e.g., Google Bard and ChatGPT) constructed by VISTEC researchers. Transformers could be divided into encoder (for source language), decoder (for target language), respectively and generator (for image generation). And the stable diffusion generator (Rombach, Blattmann, Lorenz, Esser &

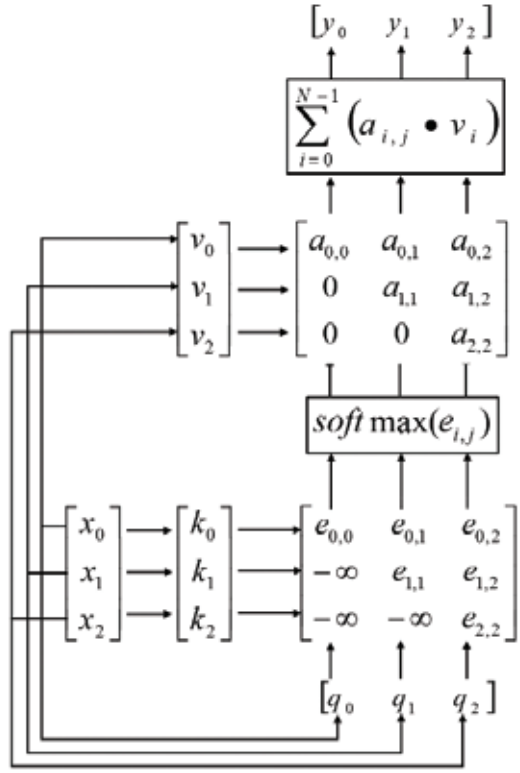


Fig. 5: Masked self-attention mechanism in decoder (in form of multi-head attention) within TH-T2I architecture.

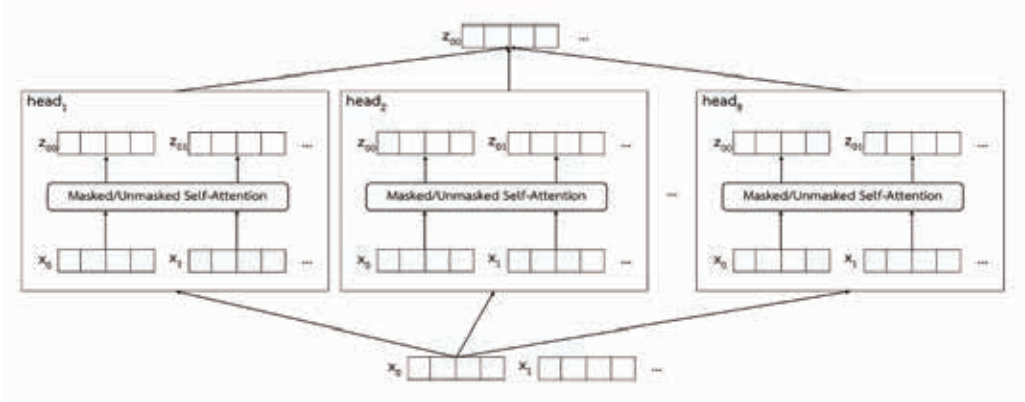


Fig. 6: Multi-head attention of masked (or unmasked) wihin TH-T2I architecture.

Ommer, 2021). was also based on a vision transformer (ViT) and used for T2I generation. The presented TH-T2I architecture was quickly shown in Figure 7. which could be further enhanced for human authentication by reCaptcha image generators (Mookdarsanit & Mookdarsanit, 2020a).

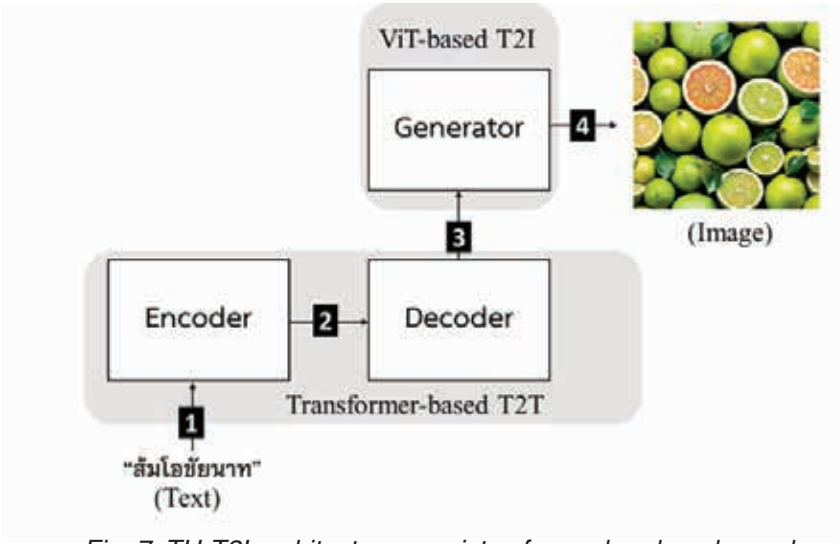


Fig. 7: TH-T2I architecture consists of encoder, decoder and generator.architecture.

3.1. Textual Transformer-based T2T Encoder

Encoder (or translation model for inputting the source language as "Thai: TH") was a language input (Lowphansirikul, Polpanumas, Rutherford & Nutanong, 2020). that might have word and phrase alignment. Encoder was measured by adequacy in (7)

$$Score_{Adequacy}(x,y) = \max\left(\mathrm{P}(\mathrm{y} = \mathrm{w}_{\text{target } i} \mid \mathrm{x} = \mathrm{w}_{\text{source}})\right) \tag{7}$$

where x (in SCB-MT-EN-TH model) as source language (Thai), y as the target language (English), and $\mathrm{P}(\mathrm{y} = \mathrm{w}_{\text{target } i} \mid \mathrm{x} = \mathrm{w}_{\text{source}})$ defined by (8)

$$\mathrm{P}(\mathrm{y} = \mathrm{w}_{\text{target } i} \mid \mathrm{x} = \mathrm{w}_{\text{source}}) = \frac{n(\mathrm{w}_{\text{target } i})}{n(\forall w_{target} \equiv \mathrm{w}_{\text{source}})} \tag{8}$$

The Transformer encoder consisted of input embedding (in part 2.1), positional embedding, multi-head self-attention (in part 2.2), skip connection, and layer normalization.

Positional embedding was the sine and cosine representation of token sequence (as well as wave frequency) that could be defined by (9), where $\varpi_k = \frac{1}{10{,}000^{2k/d}}$.

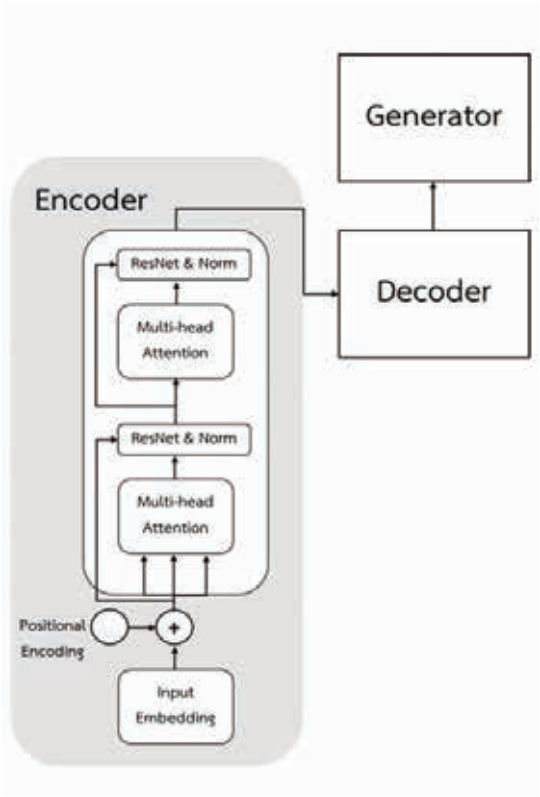


Fig. 8: Transformer-based TH-T2I encoder.

$$Positional\ \ Embedding(\mathrm{x} = \mathrm{w}_{\text{source}}) = \begin{bmatrix} \sin(\varpi_1 t) \\ \cos(\varpi_1 t) \\ \sin(\varpi_2 t) \\ \cos(\varpi_2 t) \\ \vdots \\ \sin(\varpi_{d/2} t) \\ \cos(\varpi_{d/2} t) \end{bmatrix}^T \tag{9}$$

Skip connection (or Res-block) was first proposed in the Residual network (ResNet) in 2015. ResNet won the ImageNet Large Scale Visual Recognition Challenge 2015 (ILSVRC 2015). As ResNet was the most popular pre-trained model in computer vision, it has become the baseline image classification model.

Skip connection coupled with element-wise addition and normalization with ReLU activation for pixels in image classification were proposed.

In Transformer, the layer normalization (LayerNorm) was used for tokens in the

Skip connection (or Res-block) was first proposed in the Residual network (ResNet) in 2015. ResNet won the ImageNet Large Scale Visual Recognition Challenge 2015 (ILSVRC 2015). As ResNet was the most popular pre-trained model in computer vision, it has become the baseline image classification model.

Skip connection coupled with element-wise addition and normalization with ReLU activation for pixels in image classification were proposed.

In Transformer, the layer normalization (LayerNorm) was used for tokens in the Seq2Seq model that could be defined in (10), where and $\sigma = \sqrt{\frac{1}{H}\sum_{i=1}^{H}(x-\mu)^2}$.

$$\hat{x} = \frac{x-\mu}{\sqrt{\sigma^2+\varepsilon}}\Theta\gamma + \beta \tag{10}$$

3.2. Textual transformer-based T2T decoder

The Decoder (or language model for output of the target language as "English: EN") was translated into the language output (as well as the n-Gram model with Beam Search in SMT). Decoder was measured by fluency in (11)

$$Score_{Fluency}(y) = \max\left(\sum_{j=1}^{m}\log\left(\mathrm{P}(y_j \mid < start >, \mathrm{y}_1, \mathrm{y}_2, \mathrm{y}_3, ..., \mathrm{y}_n)\right)\right) \tag{11}$$

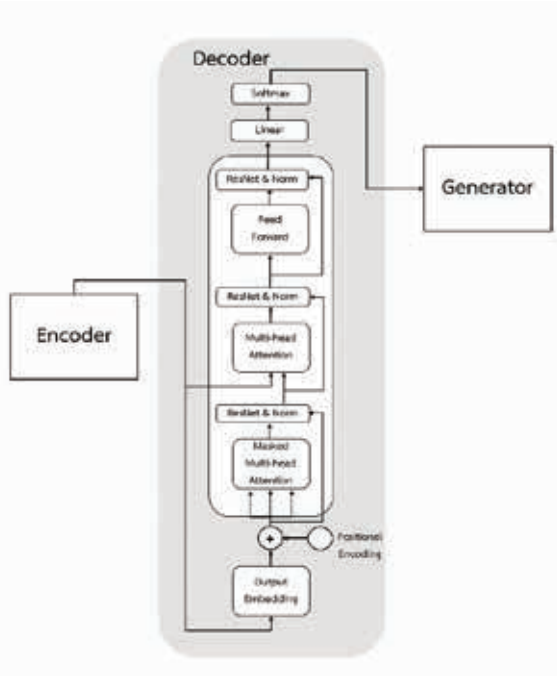


Fig. 9: Transformer-based TH-T2I decoder.

where y (in SCB-MT-EN-TH model) as the target language (English) for output of translation and $\mathrm{P}(y_j |< start >, \mathrm{y}_1, \mathrm{y}_2, \mathrm{y}_3, ..., \mathrm{y}_n)$ defined by (12)

$$\mathrm{P}(y_j |< start >, \mathrm{y}_1, \mathrm{y}_2, \mathrm{y}_3, ..., \mathrm{y}_n) = \frac{\mathrm{n}(< start > \cap \mathrm{y}_1 \cap \mathrm{y}_2 \cap \mathrm{y}_3 \cap ... \cap \mathrm{y}_j)}{\mathrm{n}(\mathrm{y}_j)} \quad (12)$$

The Transformer decoder (Lowphansirikul, Polpanumas, Rutherford & Nutanong, 2022). It consisted of positional embedding, multi-head masked and unmasked self-attention (in part 2.2), skip connection, layer normalization, and a neural network feed-forward with linear and softmax functions.

3.3. Vision transformer (ViT)-based T2I generator

The output from the previous T2T decoder (based on the SCB-MT-EN-TH model) was prompted as textual input. In TH-T2I, text-to-image generation could be done by a stable diffusion model (Lee, Hoover, Strobelt, Wang, Peng, Wright, Li, Park, Yang, Chau, 2023). The diffusion model was used to learn a data distribution by denoising a normal distribution in the UNet backbone from 2D convolution (2x2 CONV). The stable diffusion model applied the self-attention ($\tau_\theta(\bullet)$) mechanism as (1) (in part 2.1) for image generation (z_t) under the conditional textual language prompting (y), mathematically defined by (13).

$$L_{img\ gen} = \mathrm{E}_{\varepsilon(x),y,e\sim N(0,1),t}\left[\left\|\varepsilon - \varepsilon_\theta\left(z_t, t, \tau_\theta(y)\right)\right\|_2^2\right] \quad (13)$$

where = $\varepsilon_\theta\left(z_t, t, \bullet\right)$; t = 1,…, T to generate the suitable image (z_t)

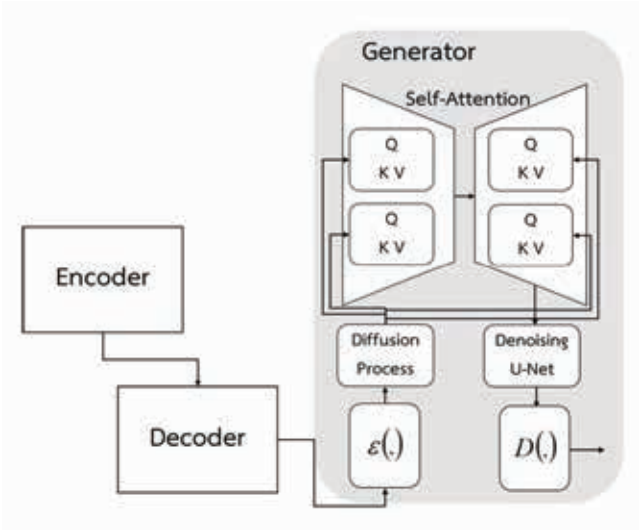


Fig.u 10: ViT-based TH-T2I generator.

Technically, a text representation generator encoded an input text prompt from the SCB-MT-EN-TH model's output to tokenize and weigh the primary significant tokens within a text for generating an image. Moreover, the image representation refiner was to generate an image in different scales and angles according to the dense vector representation until the final image representation.

4. Experimental evaluations and results

We categorize the experimental results and evaluations into two sub-parts: evaluating quality metrics of text translation and image generation and experimental T2T and T2I results.

4.*1. Evaluating metrics*

The T2T translation (as SCB-MT-EN-TH model) was evaluated by a Bi-lingual evaluation understudy (BLEU). At the same time, T2I generation (as stable diffusion) was done by Inception Score and Frechet Inception Distance (FID).

Bi-lingual evaluation understudy (BLEU) was a translation evaluation (both adequacy and fluency) by comparing machine to human translation. A higher BLEU value is better. BLEU applied precision metrics in the n-gram model (where n =1, 2, 3, 4) that could be defined by (14).

$$BLEU = \min\left(1, \frac{length_{machine}}{length_{human}}\right) \bullet \left(\prod_{i=1}^{n} precision_{i-gram}\right) \quad (14)$$

Inception score (as the name from Inception v3) was used to evaluate the quality of the generated image from the stable diffusion model (A higher score is better), defined by (15)

$$Inception = \exp\left(\mathrm{E}_{y\sim z_t} D\left(P(z \mid y) \mid P\right)(z)\right) \quad (15)$$

Frechet Inception Distance (FID) measures the distance between the generated and authentic images. (The smaller FID referred to the better quality.) The FID could be computed by (16).

$$FID = \left\|\mu(z) - \mu(z_{generated})\right\|_2^2 + Tr\left(\sum_z + \sum_{z_{generated}} - 2 \times \sqrt{\sum_z \sum_{z_{generated}}}\right) \quad (16)$$

where $\mu(z)$ and $\mu(z_{generated})$ as the average of natural images and generated images, $\|\bullet\|_2^2$ as Euclidean L2 normalization, $\sum_z$ and $\sum_{z_{generated}}$ as covariance matrices of natural images and generated images, $Tr(\bullet)$ as the main diagonal of a matrix

4.2 Experimental results

Based on 100 text prompts, the BLEU evaluation on T2T translation in Table 1; and Inception and FID evaluation on T2I generation could be evaluated in Table2, respectively.

The averaged 100 TH-EN parallel corpora was evaluated by BLEU in n-Gram (where n=1,2,3,4) as shown in Table 1. We also compared to other TH-EN machine

translation, e.g., Google translate, AI for Thai. The results showed that SCB-MT-EN-TH model provided the better performance than Google translate and AI for Thai.

Table 1: T2T translation comparison based on 100 text prompts

T2T metric	Google Translate	AI for Thai	SCB-MT-EN-TH
BLEU-1	0.57	0.61	0.67
BLEU-2	0.45	0.48	0.54
BLEU-3	0.39	0.32	0.41
BLEU-4	0.22	0.17	0.28

In case of T2I evaluation, Inception and FID were used to compare stable diffusion to other T2I generation, e.g., diffusion probabilistic model (DPM), generative adversarial network (GAN), based on 100 text prompts in term of quality of generated images in Table 2. From the experiments, T2I by stable diffusion could generate images in the highest quality as the stable diffusion applied self-attention to weight the tokens.

Table 2: T2I generation comparison based on 100 text prompts

T2I metric	DPM	GAN	Stable diffusion
Inception	1.36	1.81	2.73
FID	236.81	193.64	124.23

Some examples of Thai Text-to-Image generation shown in Figure 11, the presented TH-T2I could be contributed to many local arts and cultures to preserve Thai as the digital heritage on the open-source world, e.g., Thai amulet (Mookdarsanit, 2020)., colorful Siamese fighting fishes (Mookdarsanit & Mookdarsanit, 2019a)., Thai dance gestures (Mookdarsanit & Mookdarsanit, 2018b)., Muay-Thai Folklores (Mookdarsanit & Mookdarsanit, 2018a). or nutrients and calories estimation in Thai-foods (Mookdarsanit & Mookdarsanit, 2020c). The next Thai generation of AGI researchers could inherit these resources and materials to discover new knowledge.

Conclusion

Unlike English, Thai was a low-resource language for NLP. Thai speech, handwriting, or comments over social media could be a fruitful material and resource for growing Thai-NLP among AGI era, as well as English. There were so many gaps in Thai-NLP for Thai AGI researchers to preserve Thai as the digital heritage on the open-source world. In this paper, we propose a novel TH-T2I as a new research area of Thai-NLP to generate an image according to Thai-text prompt. Previous Transformer-based LLM researches on Thai were only T2T. This paper firstly introduce T2I in Thai. The presented TH-T2I as planting a T2I model in digital forest for Thai preservation and could be inherited by next generation of Thai AGI researchers and students. Moreover, TH-T2I could be a T2I baseline model for any local competitions (e.g., BEST Hackathon

Fig. 11: Some image generations based on Thai-text prompt engineering.

organized by NECTEC). For the shortcoming, TH-T2I was such a two-stage model (divided into T2T by SCB-MT-EN-TH and T2I by stable diffusion) that could be further developed into a single stage one.

Acknowledgement

The paper "Thai Text-to-Image Prompt Engineering by Pre-trained Large Language with Stable Diffusion Model (TH-T2I)" was presented to integrate Thai-NLP for Thai liguistic heritage conservation as Rajabhat's mission by shifting a new paradigm of Thai-NLP research area and planting TH-T2I in digital forest. Furthermore, TH-T2I could be a Thai-NLP resources, local linguistic data and programming problems. The working resources were dedicated to Chandrakasem Rajabhat University, Bangkok, Thailand.

References

Arreerard, R., Mander, S. & Piao, S. (2022). Survey on Thai NLP language resources and tools. In *Proceedings of the 13th Conference on Language Resources and Evaluation* (6495-6505). ACL.

Boonkwan, P. & Supnithi, T. (2017). Bidirectional deep learning of context representation for joint word segmentation and POS tagging. In *Proceedings of the 5th In-*

ternational Conference on Computer Science, Applied Mathematics and Applications (184-196). Berlin, Germany: Springer.

Emsawas, T. & Kijsirikul, B. (2016). Thai Printed Character Recognition using Long Short-Term Memory and Vertical Component Shifting. In *Proceedings of 14th Pacific Rim International Conference on Artificial Intelligence* (106-115). Phuket, Thailand: Sprinker.

Haruechaiyasak, C., Kongthon, A., Palingoon, P., & Trakultaweekoon, K. (2013). S-Sense: A sentiment analysis framework for social media sensing. In *Proceedings of the 6th International Joint Conference on Natural Language Processing* (6-13). Nagoya, Japan: The Association for Computational Linguistic

Ho, J., Jain, A., & Abbeel, P. (2020). Denoising diffusion probabilistic models, *arXiv: 2006.11239.*

Inthajakra, L., Prachyapruit, A. & Chantavanich, S. (2016). The Emergence of communication intellectual history in Sukhothai and Ayutthaya kingdom of Thailand. *Social Science Asia, 2*(4), 32-41.

Ketui, N., Theeramunkong, T. & Onsuwan, C. (2013). Thai news text summarization and its application. In *Proceedings of the 2013 International Symposium on Natural Language Processing,* Phuket, Thailand : AIAT.

Klahan, A., Pannoi, S., Uewichitrapochana, P. & Wiangsripanawan, R. (2018). Thai word safe segmentation with bounding extension for data indexing in search engine. In *Proceedings of the 14th International Conference on Computing and Information Technology (83-92).* Chiang Mai, Thailand: Springer.

Koanantakool, T., Karoonboonyanan, T. & Wutiwiwatchai, C. (2009). Computers and the Thai Language. *IEEE Annals of the History of Computing, 31(1),* 46-61.

Kobchaisawat, T., Chalidabhongse, T. H. & Satoh, S. (2020). Scene text detection with polygon offsetting and border augmentation. *Electronics, 9*(1), 117.

Lapjaturapit, T., Viriyayudhakom, K. & Theeramunkong, T. (2018). Multi-Candidate word segmentation using bi-directional LSTM neural networks. In *Proceedings of the 2018 International Conference on Embedded Systems and Intelligent Technology & International Conference on Information and Communication Technology for Embedded Systems* (1-6). Khon Kaen, Thailand: IEEE

Lee, S., Hoover, B., Strobelt, H., Wang, Z. J., Peng, S. Y., Wright, A., Li, K., Park, H., Yang, H. & Chau, D. H. (2023). Diffusion explainer: visual explanation for text-to-image stable diffusion, *arXiv: 2305.03509.*

Lowphansirikul, L., Polpanumas, C., J Rutherford, A. T. & Nutanong, S. (2020). scb-mt-en-th-2020: A Large English-Thai Parallel Corpus, *arXiv: 2007.03541.*

Lowphansirikul, L., Polpanumas, C., J Rutherford, A. T. & Nutanong, S. (2022). A large English–Thai parallel corpus from the web and machine-generated text. *Language Resources and Evaluation. 56*(2), 477-499.

Lowphansirikul, L., Polpanumas, C., Jantrakulchai, N. & Nutanong, S. (2021). WangchanBERTa: Pretraining transformer-based Thai Language Models, *arXiv: 2101.09635.*

Mookdarsanit, L. & Mookdarsanit, P. (2019a). SiamFishNet: The deep investigation of Siamese fighting fishes. *International Journal of Applied Computer Technology and Information Systems, 8*(2), 40-46.

Mookdarsanit, L. & Mookdarsanit, P. (2019b). Thai herb identification with medicinal properties using convolutional neural network. *Suan Sunandha Science and Technology Journal, 6(2),* 34-40.

Mookdarsanit, L. & Mookdarsanit, P. (2020a). An adversarial perturbation technique against reCaptcha image attacks. *Journal of Science and Technology Buriram Rajabhat University, 4(1),* 33-45.

Mookdarsanit, L. & Mookdarsanit, P. (2020b). The insights in computer literacy toward HR intelligence: some associative patterns between IT subjects and job positions. *Journal of Science and Technology RMUTSB, 4(2),* 12-23 .

Mookdarsanit, L. & Mookdarsanit, P. (2021a). Combating the hate speech in Thai textual memes. *Indonesian Journal of Electrical Engineering and Computer Science, 21(3),* 1493-1502.

Mookdarsanit, L. & Mookdarsanit, P. (2021b). ThaiWritableGAN: Handwriting generation under given information. *International Journal of Computing and Digital Systems, 10(1),* 689-699.

Mookdarsanit, L. & Mookdarsanit, P. (2022). Thai NLP-based Text Classification of the 21st-century Skills toward Educational Curriculum and Project Design. *International Journal of Applied Computer Technology and Information Systems, 11(2),* 62-67.

Mookdarsanit, L. & Mookdarsanit, P. (2023). The cosmetic surgery recommendation: Facial acne localization and recognition. *International Journal of Applied Computer Technology and Information Systems, 12(2),* 1-6.

Mookdarsanit, L. (2020). The intelligent genuine validation beyond online Buddhist amulet market. *International Journal of Applied Computer Technology and Information Systems, 9(2),*7-11.

Mookdarsanit, P. & Mookdarsanit, L. (2018a). A content-based image retrieval of Muay-Thai folklores by salient region matching. *International Journal of Applied Computer Technology and Information Systems, 7(2),* 21-26.

Mookdarsanit, P. & Mookdarsanit, L. (2018b). An automatic image tagging of Thai dance's gestures. In *Proceedings of Joint Conference on ACTIS & NCOBA* (76-80). Ayutthaya, Thailand.

Mookdarsanit, P. & Mookdarsanit, L. (2018c). Contextual image classification towards metadata annotation of Thai-tourist attractions. *ITMSoc Transactions on Information Technology Management, 3(1),* 32-40.

Mookdarsanit, P. & Mookdarsanit, L. (2018d). Name and recipe estimation of Thai-desserts beyond image tagging. *Kasem Bundit Engineering Journal, 8(Special Issue),* 193-203.

Mookdarsanit, P. & Mookdarsanit, L. (2019). TGF-GRU: A cyber-bullying autonomous detector of lexical Thai across social media. *NKRAFA Journal of Science and*

Technology, 15, 50-58.

Mookdarsanit, P. & Mookdarsanit, L. (2020a). Thai-IC: Thai image captioning based on CNN-RNN architecture. *International Journal of Applied Computer Technology and Information Systems, 10(1),* 40-45.

Mookdarsanit, P. & Mookdarsanit, L. (2020b). ThaiWrittenNet: Thai handwritten script recognition using deep neural networks. *Azerbaijan Journal of High Performance Computing, 3(1),* 75-93.

Mookdarsanit, P. & Mookdarsanit, L. (2020c). The autonomous nutrient and calorie analytics from a Thai food image. *Journal of Faculty Home Economics Technology RMUTP, 2(1),* 1-12.

Mookdarsanit, P. & Mookdarsanit, L. (2021a). PhosopNet: An improved grain localization and classification by image augmentation. *TELKOMNIKA Telecommunication, Computing, Electronics and Control, 19(2),* 479-490.

Mookdarsanit, P. & Mookdarsanit, L. (2021b). The COVID-19 fake news detection in Thai social texts. *Bulletin of Electrical Engineering and Informatics, 10(2),* 988-998.

Mookdarsanit, P. & Rattanasiriwongwut, M. (2017a). GPS determination of Thai-temple arts from a aingle photo. In *Proceedings of 11th International Conference on Applied Computer Technology and Information Systems (42-47).* Bangkok, Thailand.

Mookdarsanit, P. & Rattanasiriwongwut, M. (2017b). Location estimation of a photo: a Geo-signature MapReduce workflow. *Engineering Journal, 21(3),* 295-308.

Mookdarsanit, P. & Rattanasiriwongwut, M. (2017c). MONTEAN Framework: a magnificent outstanding native-Thai and ecclesiastical art network. *International Journal of Applied Computer Technology and Information Systems, 6(2),* 17-22.

Rombach, R., Blattmann, A., Lorenz, D., Esser, P. & Ommer, B. (2021). High-resolution image synthesis with latent diffusion models, *arXiv: 2112.10752.*

Ruangrajitpakorn, T. (2006). *An example-based machine translation: a case study of translating stock reports from thai to english* [Master's thesis, Chulalongkorn University]. Graduate School, Chulalongkorn University.

Soimart, L. & Mookdarsanit, P. (2016). Gender estimation of a portrait: Asian facial-significance framework. In *Proceedings of the 6th International Conference on Sciences and Social Sciences.* Mahasarakham, Thailand.

Soimart, L. & Mookdarsanit, P. (2017a). Ingredients estimation and recommendation of Thai-foods. *SNRU Journal of Science and Technology, 9(2),* 509-520.

Soimart, L. & Mookdarsanit, P. (2017b). Name with GPS auto-tagging of Thai-tourist attractions from an image. In *Proceedings of the 2nd Technology Innovation Management and Engineering Science International Conference* (211-217). Nakhon Pathom, Thailand.

Sornlertlamvanich, V. (2019). Natural language processing research in Thai context - A 29-year journey of Thai NLP. Retrieved from: https://www.slideshare.net/virach/nlp-historythaivirach20191025

Sriwirote, P., Thapiang, J., Timtong, V. & Rutherford, A. T. (2023). PhayaThaiBERT:

Enhancing a Pretrained Thai Language Model with Unassimilated Loanwords, *arXiv: 2311.12475.*

Sutthaluang, N. & Prakancharoen, S. (2020). Prediction and protection of car driving accident in urban zone. *International Journal of Innovation, Creativity and Change, 14(8),* 308-336.

Sutthaluang, N. (2019). An open library development for pesticide residue analytics in vegetables. *International Journal of Applied Computer Technology and Information Systems, 8(2),* 31-36.

Taerungruang, S. & Aroonmanakun, W. (2018). Constructing an academic Thai plagiarism corpus for benchmarking plagiarism detection systems. *GEMA Online Journal of Language Studies, 18(3),* 186-202 .

Tapsai, C., Unger, H. & Meesad, P. (2020). The application of Thai natural language processing. *Thai Natural Language Processing, 1,* 131-159.

Theeramunkong, T., Sornlertlamvanich, V., Tanhermhong,T. & Chinnan, W. (2000). Character cluster based Thai information retrieval. In *Proceedings of the 2000 International Workshop on Information Retrieval with Asian Languages,* (75-80), Hong Kong, China : ACM.

Tirasaroj, N. (2016). *A study of word sense discrimination in Thai using latent semantic analysis* [Doctoral dissertation, Chulalongkorn University]. Graduate School, Chulalongkorn University.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L. & Polosukhin, I. (2017). “Attention Is All You Need,” In *Proceedings of the 2017 International Conference on Neural Information Processing Systems,* (6000-6010). Long Beach, California : ACM.

Submitted 27.09.2023
Accepted 16.11.2023

Azerbaijan Journal of High Performance Computing, Vol 6, Issue 2, 2023, pp. 191-198
https://doi.org/10.32010/26166127.2023.6.2.191.198

Unlocking Educational Insights: Integrating Word2Vec Embeddings and Naive Bayes Classifier for Serious Game Data Analysis and Enhancement

Anar Mammadli

Azerbaijan State Oil and Industry University, Baku, Azerbaijan, anar.mammadli.az@asoiu.edu.az

*Correspondence: Anar Mammadli, Azerbaijan State Oil and Industry University, Baku, Azerbaijan, anar.mammadli.az@asoiu.edu.az

Abstract

This study explores the integration of Word2Vec embeddings and machine learning models to analyze and enhance serious game data. Word2Vec captures semantic relationships in textual content, while the Naive Bayes classifier extracts meaningful patterns. The approach improves understanding of linguistic nuances, contributing to the effectiveness of serious3 games in achieving educational objectives. Experimental results demonstrate the model's efficacy in uncovering hidden insights within the game data. This research provides a robust framework for optimizing serious game content and enhancing its educational impact.

Keyword: Serious Game, Artificial Intelligence, NLP, Text Categorization, Embeddings.

1. Introduction

Serious games, designed for educational and training purposes, have emerged as powerful tools to engage and motivate learners. These games leverage interactive and immersive experiences to facilitate learning in diverse domains. Understanding serious game data's underlying patterns and semantic structures is crucial for optimizing educational outcomes. This study investigates the integration of Word2Vec embeddings and Naive Bayes classifier models to analyze and enhance serious game data. This approach aims to unravel the intricate linguistic nuances embedded in serious games by transforming textual content into vector representations and extracting meaningful patterns. The exploration of such methodologies is pivotal for advancing the design and effectiveness of serious games in educational technology.

Word2Vec embeddings play a significant role in analyzing textual data within the context of Serious Games designed for educational and training purposes. These embeddings are instrumental in transforming raw textual content into numerical representations, thereby enabling extracting meaningful patterns and semantic structures. Integrating Word2Vec embeddings is a crucial aspect of optimizing educational outcomes and enhancing the overall effectiveness of serious games in educational technology.

This scientific paper explores the integration of Word2Vec embeddings and Naive

Bayes classifier as a robust methodology for analyzing and enhancing serious game data. The transformative power of Word2Vec lies in its ability to convert textual content into vector representations, unveiling intricate semantic structures within the linguistic fabric of serious games. When coupled with the Naive Bayes classifier, this fusion promises to unravel nuanced patterns, fostering a deeper understanding of how learners interact with educational content. The insights garnered through this approach can potentially revolutionize the design and effectiveness of serious games within the ever-evolving landscape of educational technology. This paper delves into the methodologies employed, the significance of Word2Vec embeddings, and the implications of using the Naive Bayes classifier, aiming to contribute novel perspectives to the intersection of linguistics, technology, and educational science.

2. Literature Review

The work introduced by (Marwa et al., 2017) reveals that, regardless of the language used, negative sampling emerges as the most efficient algorithm for Word2Vec in the context of topic segmentation. However, the choice of learning models requires careful consideration, as Continuous Bag of Words (CBOW) demonstrates higher efficiency with frequent words, while Skip-Gram excels with infrequent words. Compared to LSA and GloVe, Word2Vec and GloVe exhibit superior effectiveness, with Word2Vec showcasing the best word vector representations, particularly in a small-dimensional semantic space. In a comprehensive comparison, we establish that Word2Vec and GloVe outperform LSA in terms of effectiveness for topic segmentation. Furthermore, work demonstrates that Word2Vec excels over GloVe, particularly when considering the dimensionality of the semantic space.

(Pennington et al., 2014) addresses the ongoing debate between prediction-based and count-based models for word representation learning. While prediction-based models, exemplified by Baroni et al. (2014), have gained significant support, the authors argue that both classes of methods share fundamental similarities as they probe the underlying co-occurrence statistics of a corpus. The key distinction lies in the efficiency with which count-based methods capture global statistics. The authors propose a novel model, GloVe, which combines the advantages of count data efficiency with the ability to capture linear substructures found in recent prediction-based methods like word2vec. GloVe, a global log-bilinear regression model, is designed for unsupervised learning of word representations and demonstrates superior performance on various tasks, including word analogy, word similarity, and named entity recognition, compared to other existing models.

In this paper (Altszyler et al., 2017), the authors compare the efficacy of Skip-gram and Latent Semantic Analysis (LSA) in learning word embeddings from small text corpora. Their evaluation involves testing the models' ability to represent semantic categories in nested subsamples of a medium-sized corpus. The results indicate that Word2Vec embeddings outperform LSA when trained with medium-sized datasets (approximately 10 million words).

However, in scenarios with reduced corpus sizes, Word2Vec's performance significantly declines, making LSA a more suitable tool.

Several works related to machine learning have been done in the serious games area. Such as feature selection methods (Azam & Yao., 2011) for serious game data, classification of textual data [5], and automated NPC behavior generation (Dobrovsky et al., 2017; Serafim et al., 2017; Jeerige et al., 2019) from serious games. Classification also applied to textual datasets in these papers (Azam & Yao., 2011; Mammadli & Ismayilov., 2023; Zagal et al., 2005) using Convolutional Neural Network (CNN) and Graph Convolutional Network (GCN). Extensive investigations have been undertaken to enhance the efficacy of classification techniques across diverse applications, including their integration into serious games. Despite the acknowledged significance of classification in serious gaming scenarios, the exploration of Graph Neural Networks (GNNs) in this context remains comparatively limited.

3. Methodology

3.1. Preprocessing

The research endeavors to explore and analyze a specialized dataset comprising words intricately linked to a central theme. In this dataset, each entry features a main word surrounded by others that share various semantic relationships, such as synonyms, antonyms, similar words, and those evoking the essence of the main 3word. Furthermore, the dataset includes supplementary information, which will be excluded for the purposes of the experiment. The methodology adopted for this investigation involves employing word2vec as an initial step, followed by applying the Naive Bayes classifier. This multifaceted approach aims to unravel intricate word associations and leverage the power of neural networks to extract meaningful patterns from textual data. Through these techniques, the study seeks to gain valuable insights into the interconnected nature of words within the dataset.

The research commences with the acquisition of raw data in CSV format, which is subsequently subjected to preprocessing involving the removal of superfluous columns. The retained columns encompass the main word, five associated words, and the corresponding category. Following the extraction of this pertinent information, textual data transformation ensues, and the CBOW (Zhang et al., 2010) and Skip-Gram models are applied. It is noteworthy to highlight the distinctions between CBOW and Skip-Gram, two prominent algorithms in word embedding techniques. The choice between the two models is made judiciously, selecting the one that aligns optimally with the intrinsic characteristics of the dataset under investigation. This methodological approach is designed to enhance the understanding of intricate word associations within the context of the dataset.

Raw data contains the main word and five different words that are related to this main word. Also, we have a category that explains the general category of the main word. These categories include item, food, human, location, animal, profession, or other. Raw data is shown in Figure 1.

	word_tabu	first_word	second_word	...	fourth_word	fifth_word	category
0	QƏLƏM	KAĞIZ	QƏLƏMQABI	...	ÇANTA	KİTAB	item
1	FİL DİŞİ	HEYVAN	BOZ	...	32	ÜZV	other
2	MAŞIN	SÜRMƏK	TƏKƏR	...	YOL	NƏQLİYYAT	item
3	TELEFON	EKRAN	İNTERNET	...	TUŞ	SMS	item
4	HƏKİM	MÜAYİNƏ	ƏMƏLİYYAT	...	XƏSTƏ	PEŞƏ	profession
..	...	...	...	...	...	...	...

Fig. 1: The first five rows from the data source

This raw data should be converted to a dataset to apply word2vec and our model. Firstly we convert this data to a pandas frame to do quick operations. After that, we explored categories and analyzed them. In Fig 2, there are categories that we want to keep shown. Still, we have an imbalanced dataset.

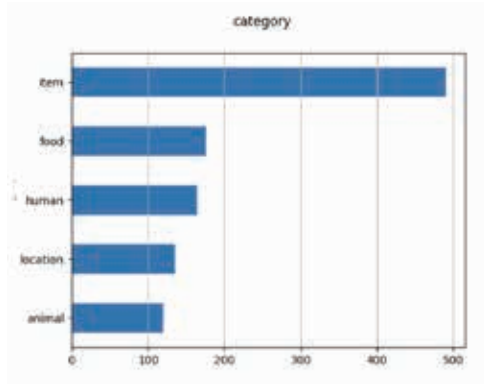


Fig. 2: Categories per number of data

Words similar to each other would be placed closer together to each other. It helps to understand how words are distributed according to each other. It helps to understand the similarity of words. You can see this in Figure 3.

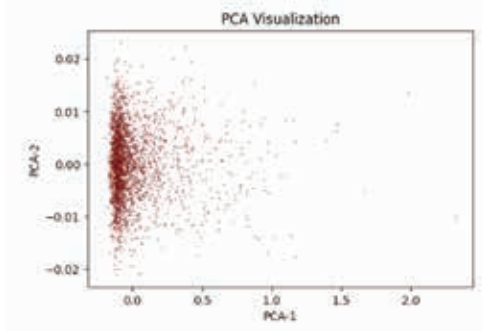


Fig. 3: PCA visualization for similarity of words

3.2. Word2Vec - Bag of Words

The bag-of-words (BOW) model is a method of converting any text into fixed-length vectors. It achieves this by tallying the frequency of each word present in the text, a procedure commonly known as vectorization. The structure closely resembles that of a feed-forward neural network. This model architecture seeks to anticipate a target word based on a set of context words. The underlying idea is straightforward: for a phrase like "Gözəl bir gün keçirin" we designate "gün" as the target word, with "gözəl," "bir," and "keçirin" as the context words. The model utilizes the distributed representations of these context words to predict the target word effectively.

3.3. Model

We employ the Naive Bayes algorithm for the model training, a sophisticated probabilistic classifier leveraging Bayes' Theorem (Vikramkumar et al., 2014). This theorem, rooted in probability theory, facilitates predictions by drawing upon prior knowledge of potentially correlated conditions. The Naive Bayes algorithm proves exceptionally apt for our dataset, evaluating each feature in isolation. It meticulously computes the probability associated with each category, culminating in the prediction of the category boasting the highest probability. This methodology, grounded in independence and precision, underscores the algorithm's suitability for our dataset's nuanced characteristics.

The classifier relies on Bayes' theorem, which is expressed as:

$$(C|X) = \frac{P(X|Ck)\cdot P(Ck)}{P(X)}$$

Where:

P(Ck|X) is the posterior probability of class Ck given the features X,

P(X|Ck) is the likelihood of observing the features X given class Ck,

P(Ck) is the prior probability of class Ck,

P(X) is the probability of observing the features X.

Now, applying the "naive" assumption of feature independence, the likelihood term can be expressed as the product of the individual feature probabilities:

$$P(X|Ck) = P(x1|Ck) \cdot P(x2|Ck) \cdot \ldots \cdot P(xn|Ck)$$

where x1, x2, …, xn are the individual features.

Assuming a document classification scenario with features representing terms or words, we can rewrite this as:

$$P(X|Ck) = P(w1|Ck) \cdot P(w2|Ck) \cdot \ldots \cdot P(wn|Ck)$$

Where w1, w2, …, wn are the terms in the document.

The classifier assigns a document to the class that maximizes the posterior probability, which can be expressed as:

$$\hat{y} = \text{argmax}_k P(Ck|X)$$

In practice, it is expected to work with logarithmic probabilities due to computational convenience and avoiding numerical underflow issues. Therefore, the decision rule becomes:

$$\hat{y} = \text{argmax}_k \log(P(Ck|X))$$

This involves computing the log-likelihoods of each term for each class and adding

them up with the logarithm of the prior probability for each class.

In summary, the Naive Bayes classifier employs Bayes' theorem, assuming feature independence, to calculate the probability of a document belonging to a particular class. The class with the highest probability is then assigned as the predicted class.

4. Results

The evaluation involves assessing its performance using various metrics. These metrics include accuracy, which measures the proportion of correct predictions; a confusion matrix providing a breakdown of correct and incorrect predictions by class; and recall, measuring the fraction of relevant instances that were successfully retrieved from the total amount. Accuracy, Precision, Recall, F1 score, and Support were used as evaluation criteria. F1-score was taken as the primary evaluation criterion since the training set is very imbalanced.

Experiments

For experiments, Serious Game data was used. This dataset consists of 1000 rows and 5 columns. The dataset is split into 70% train and 30% test.

Table 1.

	precision	recall	f1-score	support
Animal	1.00	0.55	0.71	33
Food	0.84	0.70	0.76	53
Human	1.00	0.16	0.27	51
Item	0.58	0.98	0.72	148
Location	1.00	0.10	0.18	41
Macro average	0.88	0.50	0.53	326
Weighted average	0.78	0.65	0.59	326
Accuracy			0.65	326

The Bag of Words (BoW) model got 65% of the test set right but struggles to recognize categories other than Item. The dataset we used is a small part of serious game data we used which is why it is hard to predict some categories. The confusion matrix in Figure 4. depicts that 145 out of 148 item categories are found correctly, but in other categories, it is not that precise. In our dataset, many items are related to humans, so it is hard to predict. It means that in the human category, there can be the main word human, which has an item that explains that humans and our model cannot understand that.

Conclusion and Future Works

In the results section, it is imperative to emphasize the necessity of further exploration and refinement of our dataset. A scientific approach dictates that we meticulously

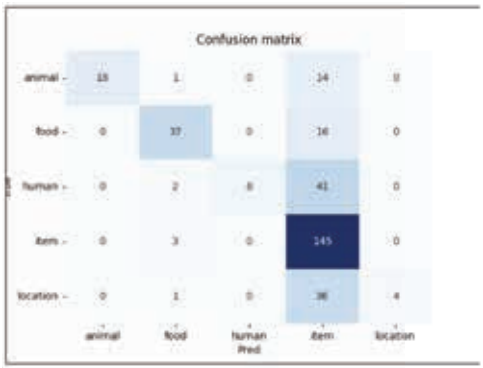


Fig. 4: Confusion Matrix

scrutinize the dataset, identifying potential areas for improvement and optimization. To enhance the robustness of our findings, we propose systematically experimenting with various models. By rigorously testing and comparing different models, we can gain insights into their strengths and weaknesses. This methodical exploration ensures that our conclusions are grounded in a comprehensive understanding of the dataset, ultimately contributing to the credibility and reliability of our study. The results underscore the potential for refinement and enhancement in our experimental approach. This suggests an opportunity for further investigation and improvement in the ongoing experiment. By identifying areas of potential optimization, we can iteratively fine-tune our methodology to yield more robust and reliable outcomes. This acknowledgment of room for improvement serves as an invitation to delve deeper into the experiment, fine-tune variables, and explore alternative avenues that may lead to heightened accuracy and efficiency in our model. This commitment to continuous improvement aligns with the dynamic nature of scientific inquiry and paves the way for a more comprehensive and impactful study.

References

Altszyler, E., Sigman, M., Ribeiro, S., & Slezak, D. F. (2016). Comparative study of LSA vs Word2vec embeddings in small corpora: a case study in dreams database. *arXiv preprint arXiv:1610.01520.*

Azam, N., & Yao, J. (2011, June). Incorporating game theory in feature selection for text categorization. In *International Workshop on Rough Sets, Fuzzy Sets, Data Mining, and Granular-Soft Computing* (pp. 215-222). Berlin, Heidelberg: Springer Berlin Heidelberg.

Bayes, T. (1968). Naive bayes classifier. *Article Sources and Contributors,* 1-9.

Dobrovsky, A., Borghoff, U. M., & Hofmann, M. (2017). Applying and augmenting deep reinforcement learning in serious games through interaction. *Periodica Polytechnica Electrical Engineering and Computer Science, 61*(2), 198-208.

Georgios N.. Yannakakis, & Togelius, J. (2018). *Artificial Intelligence and Games.*

Springer.

Jeerige, A., Bein, D., & Verma, A. (2019, January). Comparison of deep reinforcement learning approaches for intelligent game playing. In *2019 IEEE 9th Annual Computing and Communication Workshop and Conference (CCWC)* (pp. 0366-0371). IEEE.

Mammadli, A., & Ismayilov, E. A. (2023, August). Application of Deep Learning Technologies in Serious Games. In *2023 5th International Conference on Problems of Cybernetics and Informatics (PCI)* (pp. 1-4). IEEE.

Naili, M., Chaibi, A. H., & Ghezala, H. H. B. (2017). Comparative study of word embedding methods in topic segmentation. *Procedia computer science, 112,* 340-349.

Pennington, J., Socher, R., & Manning, C. D. (2014, October). Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)* (pp. 1532-1543).

Serafim, P. B. S., Nogueira, Y. L. B., Vidal, C., & Cavalcante-Neto, J. (2017, November). On the development of an autonomous agent for a 3d first-person shooter game using deep reinforcement learning. In *2017 16th Brazilian Symposium on Computer Games and Digital Entertainment (SBGames)* (pp. 155-163). IEEE.

Zagal, J. P., Mateas, M., Fernández-Vara, C., Hochhalter, B., & Lichti, N. (2005, June). Towards an ontological language for game analysis. In *DiGRA Conference* (pp. 1-13).

Zhang, Y., Jin, R., & Zhou, Z. H. (2010). Understanding bag-of-words model: a statistical framework. *International journal of machine learning and cybernetics, 1,* 43-52.

Submitted 28.09.2023
Accepted 21.11.2023

Azerbaijan Journal of High Performance Computing, Vol 6, Issue 2, 2023, pp. 199-202
https://doi.org/10.32010/26166127.2023.6.2.199.202

Predictive Modeling of Click-Through Rates: A Regression Analysis Approach

Suleyman Suleymanzade

Institute of Information Technology, Baku, Azerbaijan, suleyman.suleymanzade.nicat@gmail.com

*Correspondence: Suleyman Suleymanzade, Institute of Information Technology, Baku, Azerbaijan, suleyman.suleymanzade.nicat@gmail.com

Abstract

This research uses advanced regression techniques to develop a robust predictive model for Click-Through Rates (CTR) in online advertising. The study leverages a diverse dataset encompassing various advertising campaigns and user interactions to uncover patterns and relationships influencing click-through behavior. The goal is to provide advertisers with a tool for accurate CTR prediction, enabling them to optimize campaigns and allocate resources effectively.

Keyword: Data Splitting, XGBoost, CTR-related, CTR Prediction.

1. Introduction

The research methodology involves the application of multiple regression models, including linear regression, logistic regression, and potentially more sophisticated machine learning algorithms. Feature engineering extracts relevant information from the dataset, encompassing factors such as ad content, placement, user demographics, and contextual variables. Model performance is assessed through rigorous evaluation metrics, ensuring the reliability and generalizability of the proposed predictive framework.

The findings of this study aim to contribute valuable insights into the nuanced dynamics of CTR, shedding light on the most influential factors driving user engagement (Yang, Y., & Zhai, P., 2022; Richardson, M., Dominowska, E., & Ragno, R., 2007, May). Additionally, the research addresses the challenges associated with click fraud, emphasizing the importance of incorporating robust mechanisms to mitigate its impact on regression model accuracy.

The implications of this research extend to the practical realm of online advertising, where advertisers and marketers can leverage the developed regression model to optimize their campaigns, improve targeting strategies, and enhance overall advertising effectiveness. Ultimately, the study seeks to advance the understanding of CTR prediction through regression analysis, providing a foundation for more informed decision-making in the dynamic landscape of digital marketing.

2. Related Works

In this section, we introduce the CTR-related research works. Saura, J. R. (2021) conducted an extensive survey exploring various factors influencing online advertising's

influence on Click-Through Rates (CTR). The study reviews methodologies, challenges, and emerging trends, providing a comprehensive foundation for understanding the dynamics of CTR in digital marketing. Kamal, M., & Bablu, T. A. (2022) compared machine learning approaches for predicting CTR in online advertising. The research evaluates the performance of different algorithms, highlighting their strengths and weaknesses, and contributes insights to the ongoing discourse on effective CTR prediction models. Chen, T., & Guestrin, C. (2016, August) explored the temporal dynamics of CTR, investigating patterns and trends over different time intervals. The study provides valuable insights into how CTR varies over time, offering implications for marketers seeking to optimize campaign timing and frequency.

3. Experiment

For the experiment, we chose the Amazon CTR sales dataset (Zhou, G. et al., 2019, July). Data Splitting: The dataset was split into training (80%) and testing (20%) sets to evaluate model performance on unseen data. A baseline model was trained using a simple linear regression algorithm to establish a performance benchmark. We used XGBoost (Chen, T., & Guestrin, C., 2016, August) and CatBoost (Hancock, J. T., & Khoshgoftaar, T. M., 2020) for this experiment. The XGBoost algorithm was implemented and trained on the training dataset. Hyperparameters were fine-tuned using grid search, and model performance was evaluated on the testing set. The CatBoost algorithm, known for its adept handling of categorical features, was implemented and trained on the training dataset. Another advantage of the CatBoost model approach is that it handles categorical data not by One Hot encoding but by "Categorical Feature Embedding." Hyperparameters were fine-tuned, and the model was evaluated on the testing set.

4. Performance Metrics:

Model performance was assessed using key metrics, including Mean Squared Error and R-squared. These metrics serve as indicators of predictive accuracy and model fit. Feature Importance Analysis: Feature importance scores generated by XGBoost and CatBoost were analyzed to identify the most influential features impacting click-through rates. Cross-Validation: K-fold cross-validation (5-fold) was performed on both XGBoost

Table 1.

Metrics/Models	Baseline model	XGBoost	CatBoost
Root Mean Squared Error	0.3907	0.5362	0.5146
Error	0.6034	0.5792	0.5541
Mean Absolute Error	0.3265	0.1847	0.1683

and CatBoost to assess model stability and generalization.

Both advanced models, XGBoost and CatBoost, showcased improved predictive accuracy over the baseline, with CatBoost exhibiting the best overall performance. CatBoost's ability to handle categorical features, evident in its superior performance, is particularly advantageous in the context of Amazon advertising data, which often involves diverse categorical variables. While XGBoost demonstrated competitive results, the marginally higher RMSE indicates a nuanced trade-off between increased model complexity and prediction accuracy.

5. Future works

Further exploration of ensemble methods and hyperparameter tuning could enhance the performance of existing models (Kapoor, S., & Perrone, V., 2021). Investigate the impact of specific categorical features on click-through rates to inform targeted advertising strategies (Agarwal, D., Chen, B. C., & Elango, P., 2009, April). Evaluate model robustness over different periods and datasets to assess the generalizability of findings. In conclusion, this research contributes valuable insights into applying advanced regression models for predicting click-through rates in Amazon advertising, offering advertisers a pathway to more informed decision-making and improved campaign optimization.

References

Agarwal, D., Chen, B. C., & Elango, P. (2009, April). Spatio-temporal models for estimating click-through rate. In *Proceedings of the 18th International Conference on World Wide Web* (pp. 21-30).

Chen, T., & Guestrin, C. (2016, August). Xgboost: A scalable tree-boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 785-794).

Hancock, J. T., & Khoshgoftaar, T. M. (2020). CatBoost for big data: an interdisciplinary review. *Journal of big data, 7*(1), 1-45.

Kamal, M., & Bablu, T. A. (2022). Machine Learning Models for Predicting Click-through Rates on social media: Factors and Performance Analysis. *International Journal of Applied Machine Learning and Computational Intelligence, 12*(4), 1-14.

Kapoor, S., & Perrone, V. (2021). A Simple and Fast Baseline for Tuning Large XGBoost Models. *arXiv preprint arXiv:2111.06924.*

Richardson, M., Dominowska, E., & Ragno, R. (2007, May). Predicting clicks: estimating the click-through rate for new ads, in *Proceedings of the 16th international conference2 on World Wide Web* (pp. 521-530).

Saura, J. R. (2021). Using data sciences in digital marketing: Framework, methods, and performance metrics. *Journal of Innovation & Knowledge, 6*(2), 92-102.

Yang, Y., & Zhai, P. (2022). Click-through rate prediction in online advertising: A literature review. *Information Processing & Management, 59*(2), 102853.

Zhou, G. et al. (2019, July). Deep interest evolution network for click-through rate prediction. I*n Proceedings of the AAAI conference on artificial intelligence* (Vol. 33, No. 01, pp. 5941-5948).

Submitted 03.10.2023
Accepted 22.11.2023

Azerbaijan Journal of High Performance Computing, Vol 6, Issue 2, 2023, pp. 203-208
https://doi.org/10.32010/26166127.2023.6.2.203.208

Difference Between OpenHPC and HTCondor Cluster Systems: In-depth Analysis

Elviz Ismayilov

Azerbaijan State Oil and Industry University, Baku, Azerbaijan, elviz.ismailov@asoiu.edu.az

*Correspondence: Elviz Ismayilov, Azerbaijan State Oil and Industry University, Baku, Azerbaijan, elviz.ismailov@asoiu.edu.az

Abstract

The rapidly developing field of high-performance computing (HPC) requires efficient and scalable solutions to manage extensive computing loads. Although they use different approaches and architectures, OpenHPC and HTCondor are well-known platforms that meet these needs. This article thoroughly analyzes OpenHPC and HTCondor to identify their fundamental differences and work paradigms. OpenHPC is a comprehensive modular structure designed to facilitate the deployment, management, and maintenance of HPC clusters, offering a rich set of pre-integrated HPC software components. Conversely, HTCondor specializes in efficiently planning and managing resource-intensive tasks, using a unique partner selection system for dynamic resource allocation based on job requirements and resource availability. By examining aspects such as system architecture, resource management efficiency, scalability, flexibility, and the user ecosystem, this analysis sheds light on the strengths and weaknesses of each structure. The research aims to provide stakeholders in high-performance computing with the knowledge necessary to make informed decisions regarding the selection and implementation of high-performance computing management systems, ultimately aimed at optimizing the use of computing resources and optimizing research and development workflows.

Keyword: OpenHPC, HTCondor, HPC Clusters, High Performance Computing.

1. Introduction

OpenHPC is a community collaboration that started around 2015-2016 with the intention of developing a comprehensive and flexible open-source environment for high-performance computing (HPC) (Schulz, K. W., et al., 2016). Its foundation is based on providing system administrators and users of HPC clusters with an integrated and hardware-independent set of software components and tools needed to configure and manage HPC clusters. The initiative aims to make the deployment and management of HPC systems more accessible and efficient, promoting collaboration and exchange of best practices in the community.

The OpenHPC project was officially launched in November 2015 during the Supercomputing Conference (SC15). It was introduced as an open-source platform for

high-performance computing environments with initial support from various industrial, academic, and research organizations. The founding members included hardware manufacturers such as Intel and IBM and educational institutions, demonstrating broad support for the project and intentional cooperation between various sectors.

The main goal of OpenHPC is to provide a stable and flexible set of HPC software components independent of a specific hardware architecture. This includes tools for system administration, resource management, I/O services, development tools, and various scientific libraries. Thus, OpenHPC aims to:

• Reduce the complexity of building and managing the software stack of the HPC system.

• Promote collaboration and exchange of best practices and tools in the high-performance computing community.

• Provide the basis for creating a reproducible HPC software stack.

• Encourage the use of open standards and the compatibility of HPC systems.

Since its inception, OpenHPC has expanded to include a wide range of software components, from provisioning tools and resource managers to compilers, libraries, and development tools (Simmons, C., Schulz, K., & Simmel, D., 2020, July). He has significantly contributed to simplifying the deployment and management of HPC clusters, providing a valuable resource for experienced system administrators and newcomers to HPC. The impact of OpenHPC goes beyond simplification and efficiency improvement; it also promotes the adoption of high-performance computing technologies in various industries and research areas, reducing the entry barrier.

HTCondor is a specialized workload management system for computationally intensive tasks, also known as a batch scheduling system. It is designed to maximize computing resources by distributing high-performance tasks across available computing resources. This makes HTCondor particularly well-suited for environments that must manage large computing jobs or require detailed scheduling policies to allocate resources efficiently. The development and evolution of HTCondor highlight its importance in distributed computing and its contribution to scientific research and solving complex computational problems (Orejuela, V., Ramirez, Á. S., Toro, A. F., Gonzalez, A. F., & Briñez, D., 2018).

HTCondor development began in the late 1980s at the University of Wisconsin-Madison. It was initially conceived as a " Condor " project by Myron Livny, a professor at the university's Computer Science department. The project aimed to use unused CPU cycles of network workstations to perform resource-intensive tasks. This concept was innovative for its time and sought to create a "cycle absorber" that could efficiently use idle computing resources on the network.

• High-performance Computing: HTCondor was explicitly designed to support high-performance computing (HTC). Unlike high-performance computing (HPC), which focuses on completing a single task as quickly as possible, HTC focuses on efficiently completing many loosely coupled tasks over long periods.

• The candidate selection system. One of the main functions of HTCondor is a candidate

selection system that combines submitted assignments with the most appropriate computing resources based on a set of requirements specified by both the assignment and the resource owner.

• Checkpoints: HTCondor supports checkpoints for specific jobs, allowing you to pause and resume calculations. This feature is crucial for using energy-intensive computing resources, such as desktop workstations, which may not always be available.

DAGMan: Directed Acyclic Graph Manager (DAGMan) is a workflow management tool for HTCondor that allows users to define dependencies between jobs, providing orchestration of complex workflows (Orejuela, V., Ramirez, Á. S., Toro, A. F., Gonzalez, A. F., & Briñez, D., 2018).

HTCondor was developed by both the user community and the core development team at the University of Wisconsin-Madison. The project strongly emphasizes the principles of open-source code, allowing it to be freely used, modified, and distributed. This approach has created a large and active community of users and participants from academia, government, and industry.

HTCondor's influence is widespread in various fields requiring large-scale computational efforts, including high-energy physics, climate modeling, and bioinformatics. The flexibility and efficiency of different computing resources have made it an essential tool for researchers and organizations worldwide.

2. Related Work

OpenHPC and HTCondor are two different software products designed to work with high-performance computing systems (HPC), but they serve other purposes and are used in different contexts.

OpenHPC is an extensive software suite designed to simplify the deployment and management of HPC systems. It includes many tools and libraries for configuring the HPC environment, including cluster management, working with file systems, task planning, and monitoring. OpenHPC offers a modular architecture that allows users to select the most suitable components for their specific requirements.

HTCondor, on the other hand, is a distributed computing system optimized for managing high-performance computing tasks on many distributed resources. It allows you to organize computing resources into a pool, automatically spreading tasks between available nodes to maximize resource usage. HTCondor is especially suitable for tasks that require a large amount of computing and can effectively manage functions of varying complexity, including job queues, priorities, and dependencies between tasks.

• Purpose: OpenHPC provides tools and libraries to simplify the deployment and management of HPC systems, whereas HTCondor focuses on distributing and managing computing tasks in large distributed networks.

• Modularity: OpenHPC offers a modular structure that allows you to customize the HPC environment according to specific needs, while HTCondor provides efficient task allocation and management mechanisms.

• Application: HTCondor is ideal for projects requiring complex calculations on many independent nodes, while OpenHPC is focused on facilitating the deployment and management of the HPC infrastructure.

These tools can be used together within a single HPC system, where OpenHPC provides the basis for infrastructure deployment and management. HTCondor optimizes the distribution and execution of computing tasks.

Of course, the question arises here when it is right to choose which cluster. Here, you need to select each one according to its problems. OpenHPC offers a wide range of tools and libraries for deploying, managing, and supporting your HPC systems if you need a comprehensive platform for managing your HPC infrastructure. This is a good choice if you are building or upgrading an HPC cluster and need extensive tools for resource management, network, data storage, and security.

OpenHPC allows you to select specific components best suited to your system and requirements.

But For Distributed Computing Tasks: HTCondor is optimized for managing distributed computing tasks. If your project requires complex calculations that can be distributed across multiple nodes, HTCondor will offer an effective solution for task allocation, resource management, and job queues.

• To maximize resource usage: HTCondor is specifically designed to efficiently use available computing resources, automatically allocating tasks based on node availability and set priorities.

Sharing: In some cases, using OpenHPC for infrastructure management and HTCondor to optimize the distribution of computing tasks may be the best solution. This allows you to combine the advantages of both tools, providing a highly efficient and flexible HPC environment.

Architectural differences between OpenHPC and HTCondor: The architectural differences between OpenHPC and HTCondor reflect their different goals and approaches to managing high-performance computing (HPC) resources. While OpenHPC provides an integrated software stack for building and managing HPC clusters, HTCondor focuses on distributed computing and job optimization. Let's look at the key architectural differences:

• Integrated Software Stack: OpenHPC is a set of tools and libraries designed to simplify the deployment and management of HPC clusters. This includes task schedulers, monitoring tools, development libraries, and more, all of which can be configured to work in a single system.

• Modularity: The OpenHPC architecture is modular, allowing users to select and customize components according to their specific requirements. This makes it easier to scale and adapt the cluster to particular tasks.

• Standardization: OpenHPC strives to standardize installation and configuration processes to facilitate compatibility and interaction between the various components of the HPC system.

• Specialized Job Management System: HTCondor is explicitly designed to distribute

and manage computing jobs efficiently in large-scale distributed systems. It uses a unique queue and priority mechanism to optimize task execution.

• Dynamic Resource allocation: HTCondor can dynamically allocate jobs based on resource availability, job priorities, and usage policies. This ensures flexible and efficient use of computing resources.

• High scalability and flexibility: The HTCondor architecture is designed to work in scalable and heterogeneous computing environments, allowing it to manage jobs in various system configurations efficiently.

• Purpose and Focus: OpenHPC is designed to simplify the deployment and management of HPC clusters, while HTCondor focuses on optimizing the execution of specific computing tasks in distributed environments (Gavrilovska, A. et al., 2007, March).

• Component versus Specialization: OpenHPC offers a wide range of integrated components for various aspects of HPC cluster management, whereas HTCondor is a specialized system for job management (Diwan, S. M., 1999; Bockelman, B., Livny, M., Lin, B., & Prelz, F., 2021).

• Modularity and Standardization versus Flexibility and Scalability: OpenHPC facilitates the creation of standardized and modular HPC clusters, whereas HTCondor offers flexibility and scalability to optimize computational tasks in a wide range of computing environments.

Conslusion

The choice between OpenHPC and HTCondor depends on the specifics of your tasks, performance requirements, and available resources. It is essential to carefully evaluate both tools in the context of your goals and preferences.

OpenHPC is an integrated open-source software suite designed to simplify deploying and managing high-performance computing (HPC) clusters. This project provides users of the HPC community with a comprehensive solution that includes all the necessary tools for creating and maintaining HPC systems (Yang, Y., 2007).

OpenHPC facilitates the creation and management of HPC clusters by providing tools and resources that help researchers, engineers, and developers maximize the productivity and efficiency of their computing tasks.

HTCondor (formerly known as Condor) is a specialized job management system for computing clusters and distributed computing environments. It is designed to distribute and optimize computational tasks among available resources efficiently.

HTCondor is a powerful tool for scientific research and engineering projects that require complex calculations and efficient resource management in distributed computing environments.

The choice between OpenHPC and HTCondor (or a combination of them) depends on the project's specific requirements, including the computing scale, the types of tasks to be performed, and the available infrastructure (Hollowell, C., Barnett, J., Caramarcu, C., Strecker-Kellogg, W., Wong, A., & Zaytsev, A., 2017, October).

• The best in versatility and infrastructure management: OpenHPC.

• The best in distributed computing and efficiency: HTCondor.

Reference

Schulz, K. W., et al. (2016, November). Cluster computing with OpenHPC. In Inaugural HPC Systems Professionals Workshop, HPCSYSPROS16 (pp. 1-6).

Simmons, C., Schulz, K., & Simmel, D. (2020, July). Customizing OpenHPC: integrating additional software and provisioning new services, including open on-demand. In Proceedings of the Conference on Practice and Experience in Advanced Research Computing (p. 1).

Orejuela, V., Ramirez, Á. S., Toro, A. F., Gonzalez, A. F., & Briñez, D. (2018). Application for computational cluster performance tests configured in HTCONDOR. In MATEC Web of Conferences (Vol. 210, p. 04029). EDP Sciences.

Kalayci, S., Dasgupta, G., Fong, L., Ezenwoye, O., & Sadjadi, S. M. (2010, December). Distributed and Adaptive Execution of Condor DAGMan Workflows. In SEKE (pp. 587-590).

Hollowell, C., Barnett, J., Caramarcu, C., Strecker-Kellogg, W., Wong, A., & Zaytsev, A. (2017, October). Mixing HTC and HPC workloads with HTCondor and slurm. In Journal of Physics: Conference Series (Vol. 898, No. 8, p. 082014). IOP Publishing.

Gavrilovska, A., et al. (2007, March). High-performance hypervisor architectures: Virtualization in hpc systems. In Workshop on system-level virtualization for HPC (HPCVirt).

Diwan, S. M. (1999). Open HPC++: An open programming environment for high-performance distributed applications. Indiana University.

Bockelman, B., Livny, M., Lin, B., & Prelz, F. (2021). Principles, technologies, and time: The translational journey of the HTCondor-CE. Journal of Computational Science, 52, 101213.

Yang, Y. (2007). *A fault tolerance protocol for stateless parallel processing.* Temple University. [Ph.D thesis, AAI3293270].

Submitted 06.10.2023
Accepted 24.11.2023