IEEE Access
Multidisciplinary ⁝ Rapid Review ⁝ Open Access Journal

Received July 26, 2020, accepted August 22, 2020, date of publication August 25, 2020, date of current version September 9, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3019480

A Novel Framework for Recommending Data Mining Algorithm in Dynamic IoT Environment

M. ANWAR HOSSAIN[1], (Senior Member, IEEE), RAHATARA FERDOUSI[2,4], SK ALAMGIR HOSSAIN[3], MOHAMMED F. ALHAMID[1], (Member, IEEE), AND ABDULMOTALEB EL SADDIK[4], (Fellow, IEEE)

[1]Department of Software Engineering, College of Computer and Information Sciences, King Saud University, Riyadh 11543, Saudi Arabia
[2]Advanced Systems and Software Lab (ASySLab), Khulna, Khulna 9100, Bangladesh
[3]Computer Science and Engineering Discipline, Khulna University, Khulna 9208, Bangladesh
[4]School of Electrical Engineering and Computer Science, University of Ottawa, Ottawa, ON K1N 6N5, Canada

Corresponding author: M. Anwar Hossain (mahossain@ksu.edu.sa)

The authors would like to extend their sincere appreciation to the Deanship of Scientific Research at King Saud University, Riyadh, Saudi Arabia, for funding this Research Group No. (RG-049).

ABSTRACT Internet of Things (IoT) has been the driving force for many smart city applications. The huge volume of IoT data generated from these applications require efficient processing to get the insight, which poses significant difficulty. Data mining and machine learning (DM) algorithms are used to minimize such difficulty. However, it is still very challenging to select a particular DM algorithm that can process a dynamic IoT dataset based on some application-specific goals to achieve better accuracy. This paper proposes a knowledge-driven framework that considers the knowledge of datasets, available DM algorithms, and application goals to select the suitable DM algorithm for performing a target data processing task. This work considers data from cultural domain, health domain, and transportation domain in the experiment. The results show that the proposed approach dynamically selects the best-suited DM algorithms for the available datasets and target goals that exhibits satisfactory performance in obtaining accurate results compared to the existing work. The proposed approach not only provides flexibility in conducting dynamic IoT data mining tasks, but also reduces the complexity that would otherwise be necessary while adopting the traditional data mining approaches.

INDEX TERMS Internet of Things, data mining, machine learning, algorithm selection.

I. INTRODUCTION

The advancement of IoT in the last few years has promoted its adoption in diverse application areas including healthcare, transportation, industry management, and smart cities [1]–[3]. The huge volume of data from diverse domains has made data mining as an emerging field of IoT applications. The realistic application of data mining and machine learning has resulted in several DM algorithms in the last few decades [4]. Consequently, the machine learning approach, which is representing the method of data mining has been integrated with IoT to unleash insights of knowledge patterns with commonly practiced supervised, unsupervised and semi-supervised method [5]. Although it has been stated in an earlier work [6] that, machine learning techniques are not consistent with every data mining problem in terms of performance, recent approaches [4] have commonly adopted the ’trial and error’ based selection approach. This lacks systematic reasoning behind the selection of DM algorithm concerning appropriate matches of feature, objectives, datasets and their characteristics [7], [8].

The manual approach of selecting a DM algorithm generally provides suboptimal performance, which is not only inefficient but also requires significant human intervention as the algorithm is dedicated to performing for any particular dataset [8]. Therefore, the algorithmic instability of current machine learning algorithms to handle high dimensional large datasets requires mitigation [2], [9], [10].

One of the major complexities in mining sheer IoT data samples is that the datasets are commonly aggregated from heterogeneous sources at a variable time [1], [8]. In addition, the data sources are also diverse in terms of technologies and domains [10]. These properties of datasets can significantly impact the performance of the learning methods. Hence,

The associate editor coordinating the review of this manuscript and approving it for publication was Chun-Wei Tsai.

the selection of the best DM algorithm is dependent on the given dataset [7]. However, it is still challenging to determine which particular method should be used to analyze particular data to achieve better accuracy [11]. Thus, to recommend a DM algorithm dynamically for any dataset, meta-features of datasets can be a plus [12], [13]. For example, several intruder detection algorithms rely on meta-knowledge of the data. On the other hand, analysis of the problem domain is required to select learners automatically [14].

In general, although data mining uses a plenty of established mechanisms for data processing, clustering and classification; contemporary research still lacks in providing an autonomous and robust approach to select an optimal DM algorithm for the target dataset. This has statistical variation and changeable properties [8], [11], which make the dynamic DM algorithm selection problem challenging in IoT domain. This work aims to fill this gap.

The major contribution of this work is two-fold. First, a knowledge-driven framework has been proposed that introduces new mechanism to match knowledge of three key factors- datasets, goals and DM algorithms, for automatic selection of suitable DM algorithm in IoT smart city context. Second, several associative algorithms have been proposed to demonstrate the adaptive nature of the proposed framework in dynamic IoT environment. In order to demonstrate these contributions, we experimented with CityPlus smart city data [3], WISDM smart phone and smartwatch data [15], [16], Fitbit Experiment [17], and health data [18], [19] citelikelihood to confirm the heterogeneity of the datasets. The experiment has been conducted from two different perspectives. At first, the proposed model has been evaluated for selecting appropriate DM algorithms by considering different goals. Next, the experiment has focused on comparing the selection of DM algorithms along with their performance for various tasks, with some existing work. In both cases, the proposed approach performed satisfactorily in terms of DM algorithm selection and accuracy that proves the necessity of the work.

The remainder of this paper is organized as follows. Section II comments on the existing literature while the proposed method is elaborated in Section III. The analytical system complexity is presented in Section IV, which is followed by the experimental results and discussion in Section V. The conclusions and future work comes in Section VI.

II. RELATED WORK

Data mining as a process of finding meaningful information, from hidden patterns has been studied frequently by the researchers for traditional data in diverse application system [1], [10]. However, there is still a scarcity of data mining research for IoT that considers the heterogeneity and diverse properties of IoT [7], [20]. This section briefly comments on related work, which made the ground for the main idea of the paper.

The selection of DM algorithm based on the criteria or meta-knowledge of goals has been proposed by some researchers [11], [21]. In [11], authors found that defining an unambiguous goal can effectively solve the selection problem of machine learning algorithms. They offered an expert group-based criteria selection method, called optimum performance ranking, which is based on evaluation metrics like fitness function, statistical measures, and constraints during analysis. This work mainly focused on the supervised machine learning methods to conduct the empirical study with numerous experimental prove. However, as IoT is also trending toward closed-loop systems, such as cyber physical systems (CPS), the DM algorithm should also use unsupervised learning to ensure the robustness of the systems, where the processing is independent of a large training dataset [22].

Again in [11], authors found that a meta-learning based framework can resolve the problem of finding a best-fit DM algorithm to classify a particular dataset. They also mentioned a selection of mapping algorithm using meta-feature of problem, performance, feature, and algorithm space.

The authors in [7] emphasized that data characteristics have a significant impact on the performance of different classification methods. Thus, they conducted a study by considering both accuracy and time complexity to demonstrate how different characteristics of the dataset as an independent variable impact their results with decision tree algorithms. Instance spaces and feature selection have also been addressed as a criterion of selecting appropriate DM algorithms [1], [4] [6].

The works discussed above mainly focused either on mining traditional datasets for classification or on different supervised machine learning approaches that require increased human intervention. However, the work in [8] considered minimal human intervention and supervision into account in the context of big data analytics. This work described the potential research methodologies and activities comparing and describing the implementation process and tools for the development of a big data system.

In a nutshell, least attention has been given on dynamic data mining for IoT, which can benefit from the aggregation of meta-knowledge or characteristics of datasets/features, DM algorithms and problems/goals. This has led us to conduct the current research in order to mitigate the problem of dynamic data mining of heterogeneous IoT dataset.

III. PROPOSED FRAMEWORK

A. SYSTEM OVERVIEW

The key underlying idea of the proposed framework is to integrate and match dynamically the knowledge of Datasets (D), DM algorithms (A), and Goals (G) in order to select a suitable DM algorithm to process a particular dataset for a specific goal. Fig. 1 illustrates the proposed system architecture for the dynamic data mining model. The figure incorporates several functional units: extractDataKnowledge, extactGoalKnowledge, extractAlgoKnowledge, and matchKnowledge, which are described in subsequent sections. For the convenience of this study the details of data preprocessing are excluded, which can be found in several other sources [1], [5], [23].

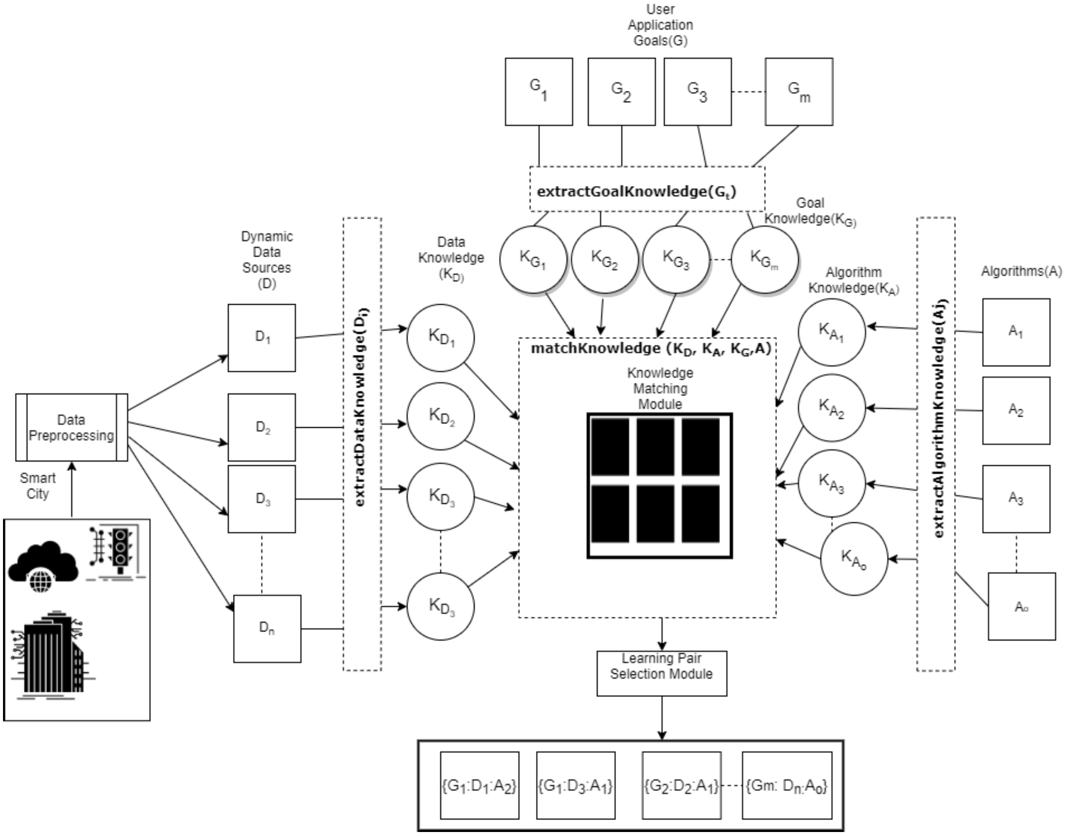


FIGURE 1. Proposed System Architecture of Dynamic Data Mining for IoT.

Algorithm 1 *DDM (D, G, A)*

```
Input: Set of Datasets D, Set of Goals G and Set of DM
       Algorithms A (supervised or unsupervised.)
Output: For each element in G for any element of D a
        potential element from A will be selected to
        obtain data mining result.
/* D, G and A will be updated at each
   execution time.                                  */
for each $D_i$ in list D do
    $K_{D_i} = extractDataKnowledge(D_i)$;
    $K_D = K_D \bigcup K_{D_i}$;
for each $G_t$ in list G do
    $K_{G_t} = extractGoalKnowledge(G_t)$;
    $K_G = K_G \bigcup K_{G_t}$;
for each $A_j$ in list A do
    $K_{A_j} = extractAlgoKnowledge(A_j)$;
    $K_A = K_A \bigcup K_{A_j}$;
$T_{pairs}$= $matchKnowledge(K_D, K_G, K_A, D, A, G)$;
/* where, $T_{pairs}$ stores the output of
   matchKnowledge                                   */
return $DM(T_{pairs})$;/* DM is general data
   mining process (classification,
   clustering or others) with the
   selected tuple $T_{pairs}$                       */
```

For the proposed system to work, a *Dynamic Data Mining (DDM)* model has been introduced (as in Algorithm 1) that takes as input D, G, and A to find the matched DM algorithm for data processing task. The methodology followed by the proposed approach is summarized below:

1) Extraction of dataset knowledge attributes using Algorithm 2, which collectively produces data knowledge, K_D (see definition in Table 1).
2) Extraction of goal knowledge attributes using Algorithm 3, which collectively produces goal knowledge, K_G (see definition in Table 1).
3) Extraction of DM algorithm knowledge attributes using Algorithm 4, which collectively produces DM algorithm knowledge, K_A (see definition in Table 1).
4) Matching among the attributes of data knowledge, goal knowledge and DM algorithm knowledge are performed using Algorithm 5 to find similarity of knowledge based on the concept of similarity checking approach introduced in [24]. This technique enables the proposed model to select appropriate DM algorithms dynamically for the heterogeneous type of data domains having multiple goals in each domain.

Here, it should be noted that the required knowledge properties are changeable in the smart-city scenario due to the heterogeneity of devices and applications. Thus,

TABLE 1. List of Parameters.

Definition of Parameters
Set of datasets, $D = \bigcup D_i$, where i = 1,2,3,...,n is an integer
Set of goals, $G = \bigcup G_t$, where t = 1,2,3,...,m is an integer and a goal G_t is an object of class *Goal*
Set of DM algorithms, $A = \bigcup A_j$, where j = 1,2,3,...,o is an integer
Set of knowledge attributes of a particular dataset, K_{D_i}
Set of knowledge attributes of a particular goal, K_{G_t}
Set of knowledge attributes of a particular DM algorithm, K_{A_j}
Collection of data knowledge, $K_D = \bigcup K_{D_i}$
Collection of goal knowledge, $K_G = \bigcup K_{G_t}$
Collection of DM algorithm knowledge, $K_A = \bigcup K_{A_j}$

Algorithm 2–Algorithm 4 include abstract functions to extract new knowledge, which in turn utilize several implemented functions in various machine learning tools such as WEKA, NumPy [23], [25].

B. DETAILS OF KNOWLEDGE EXTRACTION

This section elaborates the knowledge extraction mechanisms for dataset, goal, and DM algorithm, respectively.

1) DATA KNOWLEDGE EXTRACTION

In general, the different domains in a smart city, for example road congestion management and healthcare management, produce heterogeneous types of data that have a diverse set of properties [2]. Thus, mining data source of a particular domain becomes complex not only due to the heterogeneity of the data but also due to the variation of centroid of the data, skewness of the data, probability distribution of the data, correlation among the elements in the dataset, integrity of parameters over time, and the presence of outliers in the data [1]. For the proposed model, if we consider the data sources as D_1 = Parking Dataset, D_2 = Healthcare Dataset, etc. in a smart city, the data sources can be represented as a set of datasets D, where $D = \{D_1, D_2, D_3, \ldots, D_n\}$.

The knowledge properties of any dataset D_i are represented as K_{D_i}. The process of knowledge extraction from a dataset is given in Algorithm 2. The specific knowledge properties considered are as followings.

- Data Type: This property refers to the type of data of a dataset. For instances, binary, nominal, numerical, ordinal and mixed.
- Linearity: Linearity represents whether a dataset is linear or non-linear.
- Context: This refers to the context of the dataset. For example, diabetes, thyroid, parking, activity.

Algorithm 2 *extractDataKonwledge* (D_i)

```
Input: A realtime dataset $D_i$
Output: A list $K_{D_i}$ containing knowledge of a dataset
Initialize an empty list $K_{D_i}$
/* $getDataType, \ldots.., getDataKnowledge_x$ are $s$
   number of abstract knowledge
   retrieval functions.                            */
$K_1 = getDataType(D_i)$;
$K_3 = getLinearity(D_i)$;
$K_2 = getDataContext(D_i)$;
$K_4 = getLocation(D_i)$;
..............................
$K_s = getDataKnowledge_x(D_i)$;
$K_{D_i} = add(K_1, K_2, K_3, K_4, \ldots.K_s)$;
/* add() intregrates the knwoledge
   attibutes                                       */
return ($K_{D_i}$);
```

- Dataset Location: This property carries information about the source of the dataset. For example, Dhaka, Riyadh or any other cities.

It is worth noting that the above are some possible knowledge properties and the framework is not limited to use only these properties. Other systems, following the proposed framework can add more dynamic and unique properties of a dataset as knowledge. Thus, Algorithm 2 also uses $getDataKnowledge_x$ function, which is actually an abstract function to retrieve new data knowledge required by the system. Ultimately, the extracted knowledge of the dataset is used to find its similarity with the knowledge of goal. For example, *Data Type* property of a dataset and *goalDomainType* property of a goal can be matched to find matching dataset for a given goal, which is explained later in subsequent sections.

2) GOAL KNOWLEDGE EXTRACTION

The major challenge of DM in IoT include selecting DM algorithms for the diversity of goals for one or more data domains. A specific data domain might satisfy multiple goals. For example, a healthcare domain would require analyzing health data for the disease predictive system and providing emergency aids to patients in smart patient rooms. Goals may also differ for different datasets. For example, goals based on healthcare data are distinguishable than those based on transportation data. For the proposed model, we consider a set of goals G, where $G = \{G_1, G_2, G_3, \ldots G_t\}$ and a particular data domain D_i may need to satisfy one or more goals.

Such goal of a user application can be a data label or cluster or pattern to be predicted, which refers to general data mining tasks. Unlike knowledge of datasets, goal knowledge is almost general and predefined. Therefore, in this work a goal is defined as a class having multiple attributes, which are collectively considered as goal knowledge attributes K_{G_t} of a particular goal G_t as in the following.

Algorithm 3 *extractGoalKonwledge* (G_t)

```
Input: A particular goal $G_t$
/* $G_t$ is defined as a class,
   attributes of which will be
   defined by the system developers,
   while the values of those
   attributes are given from the
   application interface.                */
Output: A list $K_{G_t}$ containing knowledge of the goal
Initialize an empty list $K_{G_t}$
/* $getGoalProcess, \dots, getGoalKnowledge_y$ are $u$
   number of abstract knowledge
   retrieval functions.                   */
$K_1 = getGoalProcess(G_t.goalName)$;
$K_2 = getGoalDataType(G_t.goalDomainType)$;
$K_3 = getGoalModelType(G_t.goalOutputType)$;
$K_4 = getGoalTarget(G_t.goalContext)$;
$K_5 = getGoalLocation(G_t.goalCoverage)$;
................................
$K_u = getGoalKnowledge_y(G_t.newKnowledge)$;
$K_{G_t} = add(K_1, K_2, \dots, K_u)$;
return $(K_{G_t})$;
```

- *goalName*: The title of the goal (e.g. discovery or verification) represents knowledge about the objective of the DM process (e.g. prediction or hypothesis).
- *goalDomainType*: It provides knowledge about data type of target dataset (nominal, ordinal, binary, mixed) to be mined in response to the goal.
- *goalOutputType*: The output type of the DM process (e.g. class, group, number, pattern) as expected by the goal that represents knowledge about model (e.g. classification, regression, clustering, hypothesis testing).
- *goalContext*: The context of the goal (diabetes, thyroid, parking, activity) that represents knowledge about target data domain of a dataset.
- *goalCoverage*: Knowledge about location (e.g. city, country) of target data that would satisfy a goal.

The instantiation of a particular goal is done based on the values of the above attributes obtained through user-interface. The values of the goal attributes are interpreted and associated knowledge are extracted using the functions in Algorithm 3. An example of goal knowledge is, $K_{G_t} = \{K_1 =$ prediction, $K_2 =$ nominal, $K_3 =$ classification, $K_4 =$ diabetes, $K_5 =$ Paris, $\dots\}$.

However, the attributes of a goal can be updated to allow the possibility of generating more dynamic knowledge about a goal. Hence, Algorithm 3 uses *getGoalKnowledge*$_y$ function, which is an abstract function to retrieve new knowledge about a goal required by the system. The goal knowledge is later used to find the matching DM algorithm for a target dataset.

Algorithm 4 *extractAlgoKonwledge* (A_j)

```
Input: A DM algorithm $A_j$
Output: A list $K_{A_j}$ to contain DM algorithm knowledge
Initialize an empty list $K_{A_j}$
/* $getSensitivity, \dots, getAlgoKnowledge_z$ are $v$
   number of abstract knowledge
   retrieval functions.                   */
$K_1 = getSensitivity(A_j)$;
$K_3 = getExpectedDataType(A_j)$;
$K_4 = getExpectedOutputType(A_j)$;
$K_2 = getExpectedProcess(A_j)$;
................................
$K_v = getAlgoKnowledge_z(A_j)$;
$K_{A_j} = add(K_1, K_2, K_3, K_4, \dots .K_v)$;
return $(K_{A_j})$;
```

3) DM ALGORITHM KNOWLEDGE EXTRACTION

Numerous DM algorithms have been warmly accepted by the researchers [21], such as Naive Bayes (NB), Random Forest (RF), Hierarchical Clustering (HC), Featured Clustering (FC), K-Means, Decision Tree (DT), K-th Nearest Neighbor (KNN), Hidden Markov Model (HMM), Logistic Regression (LR), C4.5 DT, j48 DT, AdaBoost, Multilayer Perceptron (MLP), and support vector machine (SVM). In IoT context, the need for supervised classification or unsupervised classification (clustering) may vary for the same data domain. *For example*, mining data for predicting parking space availability may require supervised classification techniques, while finding parking location based on customers' behavior may require clustering. On the other hand, DM algorithms of similar type have different requirements that significantly vary the performance of the learning model. For example, SVM is popular for handling non-linear data points, however the computational cost is higher than the DT or KNN classifier [5]. On the contrary, despite being a HC technique, the DT has the sensitivity to all numeric datasets. Therefore, the knowledge of DM algorithm is required for the automation of IoT data mining.

The proposed model considers both supervised and unsupervised algorithms of data mining to satisfy the dynamic selection of specific DM algorithm for a particular dataset or goal. Let us consider that, A is the set of DM algorithms where,

$A = \{A_1, A_2, A_3 \dots, A_o\}$ represents numerous DM algorithms (supervised or unsupervised). The knowledge attributes K_{A_j} of any such DM algorithm A_j are extracted using Algorithm 4. Specific DM algorithm knowledge attributes are:

- Sensitivity: Represents the limitation information of a DM algorithm. In particular, whether the algorithm can accept null value, non-linear data, etc. or not.
- Expected data type: Information of data types that a DM algorithm can handle. For example, binary, nominal.

Algorithm 5 *matchKnowledge (K_D, K_A, K_G, D, A, G)*

```
Input: Knowledge of: Datasets($K_D$), Goals($K_G$), DM Algorithms($K_A$); list: goals($G$), datasets($D$), DM algorithms($A$)
Output: Return a set $T_{match}$ with matched set of tuples.
for each $G_t$ in $G$ do
    $maxSimD$, $maxSimA$ = –Infinity;/* initialize $maxSimD$ with a least value */
    $setD$, $setA$ = {};/* initialize empty set to hold matched datasets */
    for each $D_i$ in $D$ do
        $S_{G_t.D_i} = Sim(K_{G_t}, K_{D_i})$; /* similarity betw. knowledge of Goal/Dataset */
        If($S_{G_t.D_i} > maxSimD$)
        { $setD.clear()$; /* clear all previous elements from $setD$ */
        $setD.add(D_i)$; /* store datasets with highest similarity score */
        $maxSimD = S_{G_t.D_i}$;
        }
        Else If($S_{G_t.D_i} == maxSimD$)
        { $setD.add(D_i)$; }
    for each $A_j$ in $A$ do
        $S_{G_t.A_j} = Sim(K_{G_t}, K_{A_j})$ /* similarity betw. knowledge of Goal/DM Algorithm */
        If($S_{G_t.A_j} > maxSimA$)
        { $setA.clear()$;/* clear all previous elements from $setD$ */
        $setA.add(A_j)$; /* store DM algorithms with highest similarity score */
        $maxSimA = S_{G_t.A_j}$;
        }
        Else If($S_{G_t.A_j} == maxSimA$)
        { $setA.add(A_j)$; }
    /* Now, a loop will run till $e_1$ = |$setD$|, which is the cardinality of $setD$ */
    for each $D_{e_1}$ in $setD$ do
        $maxSimMerge$ = –Infinity;/* initialize $maxSimMerge$ with a least value */
        $setD\_A$ = {};/* initialize empty set to hold matched dataset and DM algorithm */
        /* Now, a loop will run till $e_2$ = |$setA$|, which is the cardinality of $setA$ */
        for each $A_{e_2}$ in $setA$ do
            /* find similarity score betw. knowledge of items in $setD$ and $setA$. */
            $S_{D_{e_1},A_{e_2}} = Sim(K_{D_{e_1}}, K_{A_{e_2}})$;
            If($S_{D_{e_1}.A_{e_2}} > maxSimMerge$)
            { $setD\_A.clear()$;/* clear all previous elements from $setD\_A$ */
            $setD\_A.add(\{D_{e_1}, A_{e_2}\})$; /* store datasets/DM algorithms pair with highest
                similarity score */
            $maxSimMerge = S_{D_{e_1}.A_{e_2}}$;
            }
        $T_{match} = T_{match} \bigcup \{G_t \times setD\_A\}$;
        /* $T_{match}$ contains selected tuples of Datasets, DM Algorithms and Goal $G_t$. */
return $T_{match}$;
```

- Expected output type: This property refers to output type of a DM algorithm. For example, class, group, and pattern.
- Expected process: Refers to the category of the DM algorithm based on data mining task. For instances, clustering, classification, and regression.

Similar to the data knowledge, the DM algorithm knowledge can also be updated. Thus, Algorithm 4 includes an abstract function *getAlgoKnowledge$_z$* to extract further DM algorithm knowledge attributes. This knowledge are matched with the knowledge of goal. For example, the *expected data type* of DM algorithm can be compared with the *expected data type* of goal, which will contribute to find the similarity between goal and DM algorithm. Similarly, DM algorithm knowledge can be compared with data knowledge to identify which of the DM algorithms should be used to process a particular dataset.

C. KNOWLEDGE MATCHING

After the knowledge extraction phases, knowledge matching process occurs using Algorithm 5. The main purpose of this algorithm is to select the matched pair of datasets and DM algorithms for the given goals based on similarity scores. After studying several similarity calculation approaches, we selected the calculation strategy mentioned in [24]. The reason behind choosing this selection is that in this work similarity between web services has been calculated considering the services as datasets and the searches used semantic and context matching, which is important for heterogeneous IoT data mining. To better understand the matching process, the workflow of Algorithm 5 has been depicted diagrammatically in Fig. 2.

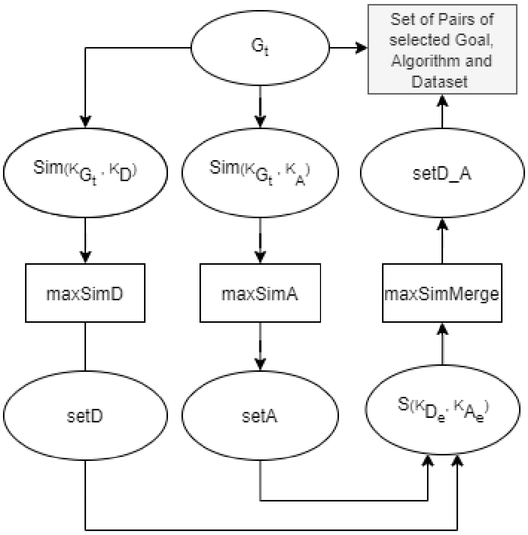


FIGURE 2. **Workflow of matching knowledge block.**

Within Algorithm 5, the outer most loop in this algorithm is for number of goals to be handled by any particular application. For example, if there are two goals G_1 and G_2, the outer most loop will run twice. In each loop, the first matching attempt calculates the maximum similarity score *maxSimD* comparing the knowledge of a particular goal and all the datasets in D. Then the selected datasets will be stored in *setD*. Similarly, maximum similarity score *maxSimA* will be calculated comparing the knowledge of a particular goal and all the DM Algorithms in A to select the candidate DM algorithms *setA*.

The same loop continues for another matching operation between the selected datasets in *setD* and selected DM algorithms in *setA*, and accordingly a pair of matching datasets and DM algorithms will be stored in *setD_A*. For example, if maximum similarity score is found between $D1$ and A_5 then $\{D_1, A_5\}$ will be saved as an item of *setD_A*. Similarly, for other datasets in *setD*, matching DM algorithms are selected. Finally, the elements of *setD_A* will be joined with associated goal and will be passed as a tuple of T_{match}.

D. WALKTHROUGH OF THE MODEL

The model illustrated in Fig. 1 can be explained further by a walkthrough of the approach as the following.

- Let there are several datasets like, D_1 = Healthcare Dataset ($city_1$), D_2 = Parking Dataset, D_3 = Healthcare Dataset ($city_2$) of a Smart City S. It is notable here that D_1 and D_3 are datasets of same context but has been produced for different city. Ideally datasets can contain information of sensors (accelerometer, gyroscope), information of device (smartwatch, smartphone), local or cloud datasets (medical ontology, disease thresholds), image datasets, etc. The proposed model uses Algorithm 2 to obtain the knowledge of D_1, D_2, and D_3 as follows.
  - $K_{D_1} = \{K_1 =$ mixed, $K_2 =$ non-linear, $K_3 =$ diabates, $K_4 =$ Riyadh, $\ldots\}$
  - $K_{D_2} = \{K_1 =$ mixed, $K_2 =$ non-linear, $K_3 =$ parking, $K_4 =$ Dhaka, $\ldots\}$
  - $K_{D_3} = \{K_1 =$ nominal, $K_2 =$ linear, $K_3 =$ diabates, $K_4 =$ Dhaka, $\ldots\}$
- Set of Goals G, where each goal is defined as a class. In particular, a goal refer to general tasks performed by DM algorithms. For this example, we consider two goals: G_1 = diabetes risk prediction and G_2 = parking availability prediction. Algorithm 3 is used to extract knowledge sets K_{G_1} and K_{G_2} for G_1 and G_2, respectively as follows.
  - $K_{G_1} = \{K_1 =$ prediction, $K_2 =$ mixed, $K_3 =$ classification, $K_4 =$ diabetes, $K_5 =$ Riyadh/Dhaka$\ldots\}$
  - $K_{G_2} = \{K_1 =$ prediction, $K_2 =$ mixed, $K_3 =$ clustering, $K_4 =$ parking, $K_5 =$ Dhaka,$\ldots\}$
- Set of DM Algorithms A, where an element of A can be any DM algorithm. Let us consider several of such algorithms as, A_1 = NB, A_2 = RF, A_3 = HC, A_4 = FC in list A. Now, Algorithm 4 is used to extract knowledge sets K_{A_1}, K_{A_2}, K_{A_3}, K_{A_4} of each of these DM algorithms as in the following.
  - K_{A_1} $\{K_1 =$ classification, $K_2 =$ mixed, $K_3 =$ class,$\ldots\}$
  - K_{A_2} $\{K_1 =$ classification, $K_2 =$ nominal, $K_3 =$ class,$\ldots\}$
  - K_{A_3} $\{K_1 =$ clustering, $K_2 =$ mixed, $K_3 =$ group,$\ldots\}$
  - K_{A_4} $\{K_1 =$ clustering, $K_2 =$ mixed, $K_3 =$ pattern,$\ldots\}$
- Algorithm 1 is used to accumulate knowledge sets K_D, K_G and K_A. It can be observed that there are semantic matches among the values represented by K_D, K_G and K_A, which are accomplished by Algorithm 5. First for each goal, a set of potentially matching datasets (*setD*) and DM algorithms (*setA*) are selected. Then a final matching is conducted among these two sets to pick the matching DM algorithm for each selected dataset. Let, $Sim(K_{G_1}, K_{D_1}) = Sim(K_{G_1}, K_{D_3}) = 0.87$ then $setD = \{D_1, D_3\}$ for G_1. In the same manner, assume $setA = \{A_1, A_2\}$ as per knowledge matching between G_1 and available DM algorithms. That is, NB and RF are selected for diabetes prediction. Algorithm 5 then calculates the similarity between elements of *setD* and

setA. If $Sim(K_{D_1}, K_{A_1}) > Sim(K_{D_1}, K_{A_2})$ then the tuple $\{G_1, D_1, A_1\}$ is returned as the selected set. Again, if $Sim(K_{D_3}, K_{A_2}) > Sim(K_{D_3}, K_{A_1})$ then $\{G_1, D_3, A_2\}$ is returned for the case of G_1. So finally, for diabetes prediction, RF will be selected to process Healthcare Dataset($city_1$) and NB will be selected to process Healthcare Dataset($city_2$). The process repeats for finding the match for G_2.

It should be noted here that the goal, datasets, and DM algorithms can be specified in a dynamic fashion, and the proposed approach selects the best matching dataset and DM algorithm for the specified goal to satisfy a client application's need. The knowledge attributes specified for G, D, and A can also be updated, which better satisfy the dynamic context of IoT data mining.

IV. COMPLEXITY ANALYSIS

The worst-case computational cost of the proposed framework has been detailed here. The cost approximation is divided into two phases: knowledge retrieval phase and knowledge matching phase. In the Knowledge retrieval phase, *extractDataKnowledge*(), *extractGoalKnowledge*(), and *extractAlgoKnowledge*() in Algorithm 1 are evaluated, whereas for the Knowledge matching phase, *matchKnowledge*() algorithm is evaluated.

A. COMPLEXITY IN KNOWLEDGE RETRIEVAL

Taking a closer look at Algorithm 1 it can be observed that the upper bound of first for loop is n. For each call the *extractDataKnowledge*() algorithm is executed to extract s knowledge items for each dataset. In the dataset, different data types may exist and so the algorithm needs to perform row-wise and column-wise search. Hence, in brute-Force case, the complexity of obtaining knowledge of n number of datasets, K_D, can be approximated as $O(s.n.1) \cong O(n^2)$ when considering $s = n$.

Again, for second loop in Algorithm 1, u number of goal knowledge items for each goal is extracted using the *extractGoalKnowledge*() algorithm. So for a total m number of goals, the complexity of obtaining overall goal knowledge, K_G, can be approximated as $O(u.m.1)$ when each of $K_1, K_2, \ldots, K_u$ takes 1 CPU cycle. Now, considering $u = m$, the worst case complexity of goal knowledge extraction becomes $O(m^2)$.

Similarly, the complexity to extract the knowledge of data mining algorithms, K_A, is $O(v.o.1) = O(o^2)$ when $v = o$. Now, if n, m, and o are close proximity values, the overall complexity of knowledge retrieval phase approximates to $O(n^2) + O(m^2) + O(o^2) \cong O(n^2)$.

B. COMPLEXITY IN KNOWLEDGE MATCHING

After the knowledge retrieval process, the knowledge matching process starts in Algorithm 1 by executing the *matchKnowledge*() function, where the outer loop runs for m times. Then two matching operations are performed: matching between goal knowledge and dataset knowledge to get *setD*; matching between goal knowledge and algorithm knowledge to get *setA*. If we use a better search algorithm for matching, then the complexity to obtain *setD* is $O(nlog\ n)$ for n number of datasets. Likewise, the complexity to obtain *setA* is $O(olog\ o)$ for o number of algorithms. Together, for m number of goals, the complexity to obtain *setD* and *setA* becomes $O(mnlog\ n) + O(molog\ o)$.

Now, the remaining loops in Algorithm 1 are carried out to obtain *setD_A* by matching the knowledge between each of the e_1 number of selected datasets in *setD* and e_2 number of selected algorithms in *setA* for m goals; the complexity of which is approximated to $O(m.e_1.e_2log\ e_2)$.

Thus, the overall complexity of knowledge matching phase can be approximated to $O(mnlog\ n) + O(molog\ o) + O(m.e_1.e_2log\ e_2)$. As e_1 and e_2 are very small value, the portion $O(m.e_1.e_2log\ e_2)$ would be negligible. Now assuming, m, n, and o are close proximity values, the overall knowledge matching complexity results in $O(n^2log\ n)$, which is very reasonable.

V. EXPERIMENTAL ANALYSIS

This section presents experimental detail based on the proposed approach. Accordingly, the experimental setup for this study is elaborated. Next, it describes the two experiments that were carried out to obtain the results, which is followed by a discussion of the findings to justify the novelty of the work. The two experiments are:

- Performance analysis of the proposed framework with different goals (data mining tasks).
- Comparison with existing work for the different data mining tasks.

A. EXPERIMENTAL SETUP

In order to conduct the experiment, we considered a set of application goals, different datasets, and available DM algorithms. The datasets are collected from different domains to satisfy the heterogeneity of data sources in the context of IoT [3], [15] [16], [17] [26]. There are many DM algorithms as well as about 179 distinct ways for implementing supervised algorithms [8]. As this paper specifically focuses on evaluating the mechanisms proposed in the framework itself, the existing available implementations of DM algorithms from the Python Libraries [23] are considered here. The datasets have been pre-processed to eliminate the null values and missing values by assigning mode value and by ignoring missing tuples, respectively. For the first experiment, some general goals in both supervised and unsupervised problem types are considered. As for the second experiment, several goals aligned with existing work are considered.

B. EXPERIMENT-1: PERFORMANCE ANALYSIS OF THE PROPOSED FRAMEWORK

In this experiment, the following goals for a dataset of cultural events are considered.

- G_1: Community recommendation with similar recreation taste.
- G_2: Community recommendation with similar music choice.
- G_3: Favorite music prediction.
- G_4: Favorite shop prediction.
- G_5: Parking availability prediction.

The City Pulse smart city data source has been used as the main data source. This data source contains various types of datasets produced from different smart city scenarios [3].

In the case of experiment-1, multiple datasets from the City Pulse data source that are available in CSV format have been used. The details of those datasets are tabulated in Table 2. It can be observed here that there are some datasets like Parking Data-1, Parking data-2, which represent data of the same domain but are stored in different datasets. This scenario has been considered while designing the proposed algorithms. Besides, multiple goals like G_1-G_4 are satisfied using the same datasets DS06 and DS07.

TABLE 2. Detail of datasets used from CityPlus data source.

Dataset ID	Dataset name	Number of instances	Type of data
D01	Road Traffic Data-1	9306	Real
D02	Road Traffic Data-2	9472	Real
D03	Road Traffic Data-3	10000	Real
D04	Road Traffic Data-4	9597	Real
D05	Pollution Data	17569	Generated
D06	Cultural Event -1	100	Real
D07	Cultural Event-2	100	Real
D08	Parking Data-1	55265	Real
D09	Parking Data-2	55267	Real

As per the proposed framework, the matchKnowledge() algorithm selects datasets and DM algorithms for each goal after matching the knowledge of goals with the knowledge of datasets and DM algorithms, respectively. This algorithm mainly generates three sets, which contains the selection of: datasets matched with a goal (*setD*); DM algorithms matched with a goal (*setA*), and pair of datasets and DM algorithms for the given goals (*setD_A*). The outputs in these sets are given in Table 3.

It is clear that for each goal multiple datasets and multiple DM algorithms can be selected, which is shown in the third column of Table 3. For example, G_1 and G_2 are clustering problem and multiple DM clustering algorithms have been selected in *setA* as candidate. The fourth column of Table 3 represents the pair of selected dataset and DM algorithm for the respective goal. If this process was otherwise conducted manually, any of the DM algorithms could have been selected for the task. However, the proposed framework provides the selection that is the best among the selected candidate DM algorithms. This becomes evident after applying the selected DM algorithms on corresponding datasets, which results in high accuracy value of about 90%-98% as depicted in Fig. 3.

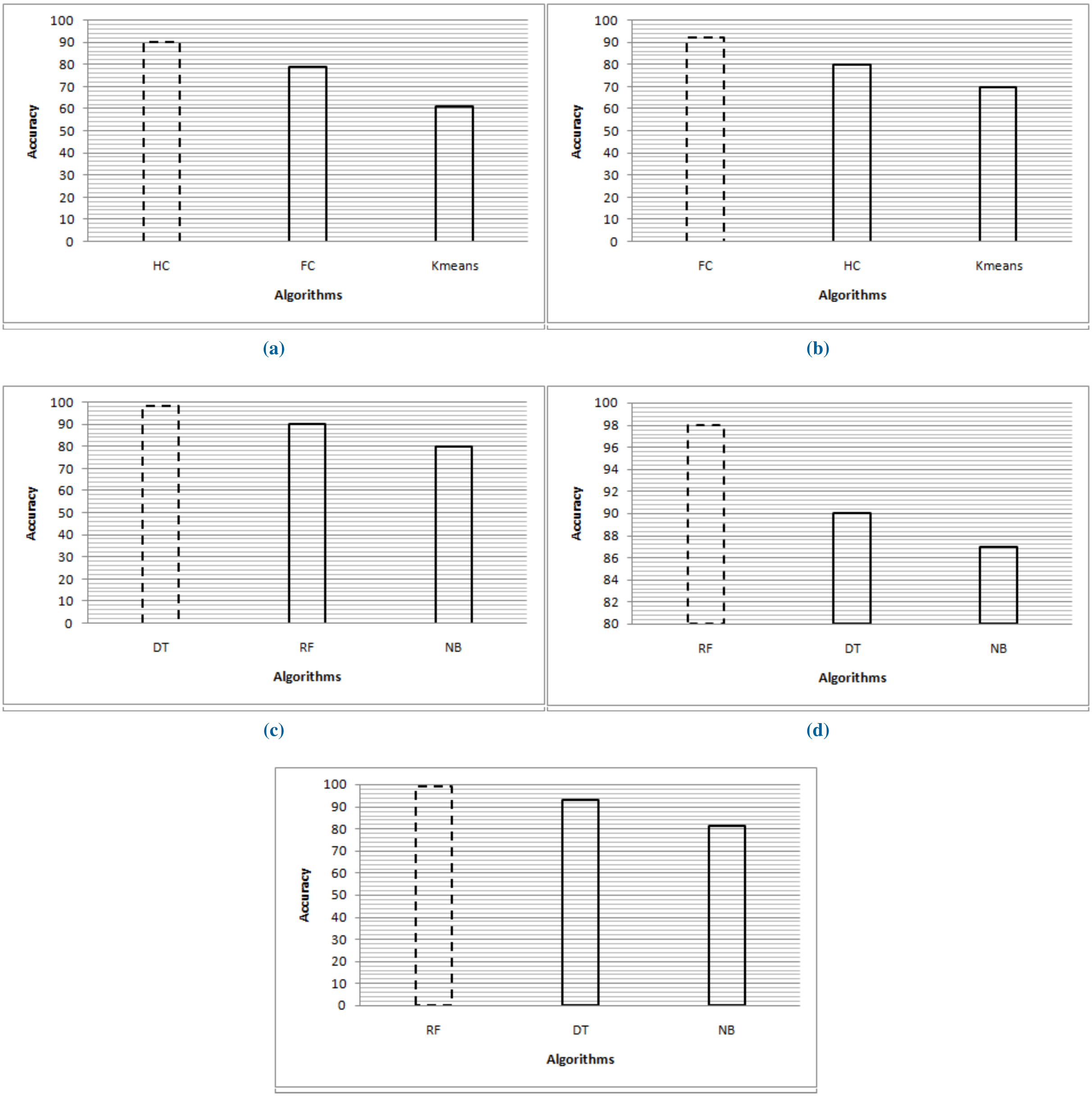
TABLE 3. Selection by matchKnowledge() Algorithm.

Goals	Output in setD	Output in setA	Output in setD_A
G1	D06 D07	HC, FC, K-Means HC, FC, K-Means	D06, HC D07, HC
G2	D06 D07	HC, FC, K-Means HC, FC, K-Means	D06, FC D07, FC
G3	D06 D07	NB, DT, RF NB, DT, RF	D06, DT D07, DT
G4	D06 D07	NB, DT, RF NB, DT, RF	D06, RF D07, RF
G5	D08 D09	NB, DT, RF NB, DT, RF	D08, RF D08, RF

C. EXPERIMENT-2: COMPARISON WITH EXISTING WORK

This experiment evaluates the proposed framework to demonstrate whether it can select the best DM algorithms among the many available DM algorithms without human intervention or not. For this purpose, some existing data mining work have been considered, which applied different DM algorithms for various target tasks. In our case, we considered similar goals and obtained the recommended DM algorithm dynamically for each case based on the proposed approach to demonstrate that the selected DM algorithm indeed produces results that are comparable to the existing approaches.

In Table 4, the second column shows the different existing work that considered different goals, such as diabetes prediction [18] and false-alarm detection [28], as well as the DM algorithms that they have used to process the data for the target goal. The third column of the table lists the DM algorithms that are recommended by the proposed method. The accuracy of the DM algorithms in both approaches for the target tasks are reported in Fig. 4. It shows that the accuracy obtained in both existing and proposed cases are comparable. However, it can be observed that in some occasions, such as thyroid prediction, the selection of DM algorithm by the proposed approach is different than the existing approach. The existing work proposes NB as the best DM algorithm with 75% accuracy for thyroid prediction, whereas the proposed framework achieves 99% accuracy with AdaBoost (ensemble technique) algorithm. Again, for the upstairs activity detection case, the existing work proposes MLP as the best DM algorithm with an accuracy of 61.5%, while the proposed framework recommends RF algorithm that achieves 70% accuracy. This experiment clearly indicates that the proposed framework is able to select appropriate DM algorithms dynamically in IoT environment, where the selected DM algorithm generates results as per the given goal and dataset.



(a)

(b)

(c)

(d)

(e)

FIGURE 3. Accuracy comparison of candidate DM algorithms in setA for each goal, where the dashed border-lined bar represents the recommended DM algorithm by the proposed framework. (a) Accuracy comparison for G_1 (b) Accuracy comparison for G_2 (c) Accuracy comparison for G_3 (d) Accuracy comparison for G_4 (e) Accuracy comparison for G_5.

D. DISCUSSION

From the above experiment, it can be stated that the proposed framework provides a sophisticated mechanism for data mining in IoT. It shows that the proposed framework capable in selecting a suitable DM algorithm for the target tasks in a dynamically changing environment. The results interestingly demonstrate the adaptable nature of the framework with diverse datasets (see Table 2), as well as its suitability for non-identical goals of different applications (see Table 4).

Despite the merit the proposed framework posses, there is limitation in the approach. One of such limitations is related to the similarity computation process between goal, dataset, and algorithm in the knowledge matching phase, where we only explored the technique presented in [24]. However, other options of similarity computation are open for exploration, which would give alternative choice for DM algorithm selection.

The proposed work has a positive implication in that it is capable of selecting appropriate DM algorithms for the data mining task, which is often facilitated by researchers or developers through numerous time-consuming procedures. These procedures in general include the study of several

TABLE 4. Comparative result of existing and the proposed work.

Goals	Proposed algorithm in existing work	Selected DM algorithm by the proposed framework
Diabetes Prediction [18]	KNN,RF	RF
Stroke Prediction [27]	C4.5	C4.5
False-Alarm Detection [28]	HMM	HMM
Thyroid Prediction [19]	NB	AdaBoost
Diabetes Risk Prediction [26]	RF	RF
Activity Detection (Walking) [15]	LR	RF
Activity Detection (Jogging) [15]	MLP	MLP
Activity Detection (Sitting) [15]	j48	DT
Activity Detection (Standing) [15]	j48	DT
Activity Detection (Upstairs) [15]	MLP	RF

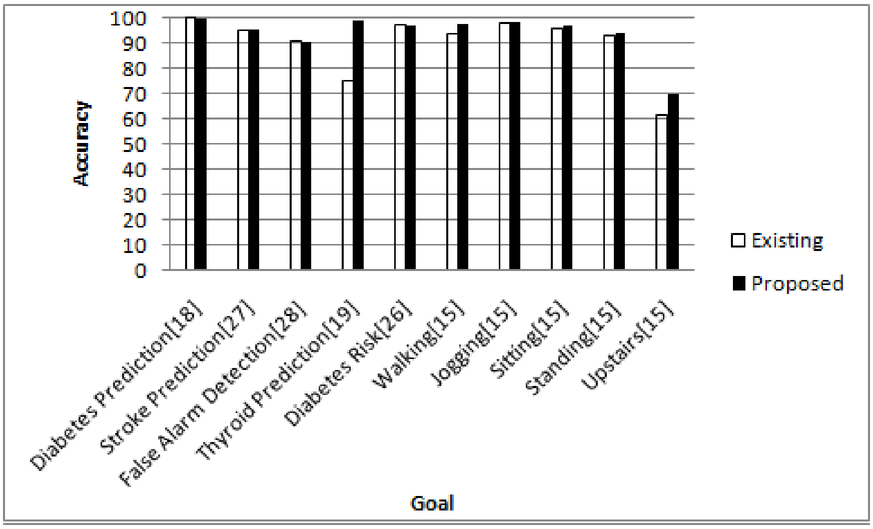


FIGURE 4. Comparative analysis between existing work and proposed framework.

existing DM algorithms repeatedly to choose the best fit for a specific dataset. Then the implementation support from the data mining tools, such as WEKA, Scikit-learn or pure python or C code are tested, which are time consuming. Undoubtedly, automatic selection of DM algorithm can contribute toward reducing both the time and cost of these additional steps to accomplish a data mining process, which is confirmed in the experiments presented in the paper. Thus, it can be argued that the proposed framework can be considered as the next frontier for data mining in a dynamic IoT environment.

VI. CONCLUSION AND FUTURE WORK

This paper focused on the challenging issue of selecting a particular DM algorithm to analyze dynamic IoT data based on the knowledge of dataset, application goals, and a set of existing DM algorithms. Consequently, a novel framework is proposed that considers such knowledge to select the best possible algorithm that would provide high accuracy given the data and application goal. We conducted experiments by considering data from health domain, transportation domain and cultural domain. As per the experiment, several DM algorithms have been selected for processing different datasets to satisfy diverse application goals. The experimental results confirmed that the DM algorithm recommended for the target goal, matches the recommendation by existing work. Besides, the accuracy of the data processing tasks based on the selected DM algorithms often surpassed the results declared in existing research.

As a future work, the proposed framework can be extended in different directions. One area of extension would be to integrate service providers to provision services within a particular application domain based on the outcome of the proposed framework in real-time. The other obvious direction would be to develop a practical application with diverse goals and study the scalability issue of the framework considering the huge volume of IoT data generated from a smart city deployment.

REFERENCES

[1] M. M. Gaber, A. Aneiba, S. Basurra, O. Batty, A. M. Elmisery, Y. Kovalchuk, and M. H. U. Rehman, "Internet of Things and data mining: From applications to techniques and systems," *Wiley Interdiscipl. Rev., Data Mining Knowl. Discovery*, vol. 9, no. 3, p. e1292, May 2019.

[2] L. Cui, S. Yang, F. Chen, Z. Ming, N. Lu, and J. Qin, "A survey on application of machine learning for Internet of Things," *Int. J. Mach. Learn. Cybern.*, vol. 9, no. 8, pp. 1399–1417, 2018.

[3] M. Giatsoglou, D. Chatzakou, V. Gkatziaki, A. Vakali, and L. Anthopoulos, "CityPulse: A platform prototype for smart city social data mining," *J. Knowl. Economy*, vol. 7, no. 2, pp. 344–372, Jun. 2016.

[4] M. A. Muñoz, L. Villanova, D. Baatar, and K. Smith-Miles, "Instance spaces for machine learning classification," *Mach. Learn.*, vol. 107, no. 1, pp. 109–147, Jan. 2018.

[5] N. Jain and V. Srivastava, "Data mining techniques: A survey paper," *Int. J. Res. Eng. Technol.*, vol. 2, no. 11, pp. 1163–2319, 2013.

[6] A. L. Blum and P. Langley, "Selection of relevant features and examples in machine learning," *Artif. Intell.*, vol. 97, nos. 1–2, pp. 245–271, Dec. 1997.

[7] D. Oreski, S. Oreski, and B. Klicek, "Effects of dataset characteristics on the performance of feature selection techniques," *Appl. Soft Comput.*, vol. 52, pp. 109–119, Mar. 2017.

[8] A. Nazir, "Seamless automation and integration of machine learning capabilities for big data analytics," *Int. J. Distrib. Parallel Syst.*, vol. 8, no. 3, pp. 1–18, 2017.

[9] S. Ali and K. A. Smith, "On learning algorithm selection for classification," *Appl. Soft Comput.*, vol. 6, no. 2, pp. 119–138, Jan. 2006.

[10] L. Li and A. Ghasemi, "IoT-enabled machine learning for an algorithmic spectrum decision process," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 1911–1919, Apr. 2019.

[11] R. Ali, S. Lee, and T. C. Chung, "Accurate multi-criteria decision making methodology for recommending machine learning algorithm," *Expert Syst. Appl.*, vol. 71, pp. 257–278, Apr. 2017.

[12] L. Kotthoff, I. P. Gent, and I. Miguel, "A preliminary evaluation of machine learning in algorithm selection for search problems," in *Proc. 4th Annu. Symp. Combinat. Search*, 2011, pp. 1–8.

[13] Y. Zhang, Y. Xin, Q. Li, J. Ma, S. Li, X. Lv, and W. Lv, "Empirical study of seven data mining algorithms on different characteristics of datasets for biomedical classification applications," *Biomed. Eng. OnLine*, vol. 16, no. 1, p. 125, Dec. 2017.

[14] T. Doan and J. Kalita, "Predicting run time of classification algorithms using meta-learning," *Int. J. Mach. Learn. Cybern.*, vol. 8, no. 6, pp. 1929–1943, Dec. 2017.

[15] G. M. Weiss, K. Yoneda, and T. Hayajneh, "Smartphone and smartwatch-based biometrics using activities of daily living," *IEEE Access*, vol. 7, pp. 133190–133202, 2019.

[16] G. M. Weiss. (Oct. 2019). *WISDM Smartphone and Smartwatch Activity and Biometrics Dataset Data Set*. UCI Machine Learning Repository. Accessed: May 10, 2020. [Online]. Available: https://archive.ics.uci.edu/ml/datasets/WISDM+Smartphone+and+Smartwatch+Activity+and+Biometrics+Dataset+

[17] K. Pearson. (Jun. 2019) *Fitbit. Data. World*. Accessed: May 30, 2020. [Online]. Available: https://data.world/kmpearson/experiment

[18] J. P. Kandhasamy and S. Balamurali, "Performance analysis of classifier models to predict diabetes mellitus," *Procedia Comput. Sci.*, vol. 47, pp. 45–51, Jan. 2015.

[19] E. Turanoglu-Bekar, G. Ulutagay, and S. Kantarcı-Savas, "Classification of thyroid disease by using data mining models: A comparison of decision tree algorithms," *Oxford J. Intell. Decis. Data Sci.*, vol. 2016, no. 2, pp. 13–28, 2016.

[20] W. Gan, J. C.-W. Lin, H.-C. Chao, A. V. Vasilakos, and P. S. Yu, "Utility-driven data analytics on uncertain data," *IEEE Syst. J.*, early access, Mar. 23, 2020, doi: 10.1109/JSYST.2020.2979279.

[21] N. Pise and P. Kulkarni, "Algorithm selection for classification problems," in *Proc. SAI Comput. Conf. (SAI)*, Jul. 2016, pp. 203–211.

[22] L.-A. Tang, J. Han, and G. Jiang, "Mining sensor data in cyber-physical systems," *Tsinghua Sci. Technol.*, vol. 19, no. 3, pp. 225–234, Jun. 2014.

[23] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, and J. Vanderplas, "Scikit-learn: Machine learning in Python," *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, Oct. 2011.

[24] T. Rachad, J. Boutahar, and S. El ghazi, "A new efficient method for calculating similarity between Web services," 2015, *arXiv:1501.05940*. [Online]. Available: http://arxiv.org/abs/1501.05940

[25] I. H. Witten, E. Frank, M. A. Hall, and C. J. Pal, *Data Mining: Practical Machine Learning Tools and Techniques*. San Mateo, CA, USA: Morgan Kaufmann, 2016.

[26] M. F. Islam, R. Ferdousi, S. Rahman, and H. Y. Bushra, "Likelihood prediction of diabetes at early stage using data mining techniques," in *Computer Vision and Machine Intelligence in Medical Image Analysis*. Singapore: Springer, 2020, pp. 113–125.

[27] L. Amini, R. Azarpazhouh, M. T. Farzadfar, S. A. Mousavi, F. Jazaieri, F. Khorvash, R. Norouzi, and N. Toghianfar, "Prediction and control of stroke by data mining," *Int. J. Preventive Med.*, vol. 4, no. 2, p. S245, 2013.

[28] M. G. Al Zamil, S. Samarah, M. Rawashdeh, M. S. Hossain, M. F. Alhamid, M. Guizani, and A. Alnusair, "False-alarm detection in the fog-based Internet of connected vehicles," *IEEE Trans. Veh. Technol.*, vol. 68, no. 7, pp. 7035–7044, Jul. 2019.

M. ANWAR HOSSAIN (Senior Member, IEEE) received the bachelor's degree in computer science and engineering from Khulna University, Bangladesh, and the master's degree in computer science and the Ph.D. degree in electrical and computer engineering from the University of Ottawa, Canada, in 2005 and 2010, respectively. He is currently an Associate Professor with the Department of Software Engineering, College of Computer and Information Sciences, King Saud University (KSU), Riyadh, Saudi Arabia. His current research includes the Internet of Things, multimedia surveillance and privacy, assisted living, artificial intelligence, and software engineering. He has authored/coauthored over 100 research articles. He has co-organized several IEEE/ACM workshops including IEEE ICME AAMS-PS 2011-13, IEEE ICME AMUSE 2014, ACM MM EMASC-2014, IEEE ISM CMAS-CITY2015, IEEE ICME MMCloudCity-2016, and IEEE ISM EMASC-2017 workshop. He is a Senior Member of ACM. He served as a Guest Editor of Springer *Multimedia Tools and Applications* journal, *International Journal of Distributed Sensor Networks*, and Springer *Multimedia Systems* journal. He is an Associate Editor in several journals. He has secured several grants for research and innovation.

RAHATARA FERDOUSI received the bachelor's degree in computer science and engineering from Khulna University, Bangladesh. She is currently a Researcher with the Advanced Systems and Software Research Lab (ASysLab), Khulna, Bangladesh. Previously, she worked as a faculty at Metropolitan University, Sylhet, Bangladesh. She has been recognized as national and international innovator by a2i Programme, Prime Minister office in Bangladesh and UN Women. She has several awards for her innovative ideas to utilize technology for human welfare. As a new researcher, she has authored/coauthored articles in reputed journals/conferences of IEEE and Springer.

SK ALAMGIR HOSSAIN received the B.Sc. (Engg.) degree in computer science and engineering from Khulna University, Khulna, Bangladesh, in 2006, and the M.C.S. degree in computer science from the University of Ottawa, Ottawa, ON, Canada, in 2011. He is currently pursuing the Ph.D. degree in computer science and engineering, Khulna University, Khulna, Bangladesh. At the University of Ottawa, he was with the Multimedia Communications Research Laboratory (MCRLab), School of Information Technology and Engineering. He is also a faculty with the Computer Science Department, Khulna University. He has authored or coauthored more than 30 publications including refereed journals, conference papers, and book chapter. His research interests include the Internet of Things (IoT), smart environment, ambient intelligence and humanized computing, and virtual reality with haptic.

MOHAMMED F. ALHAMID (Member, IEEE) received the Ph.D. degree in computer science from the University of Ottawa, Canada. He is currently an Associate Professor with the Software Engineering Department, King Saud University, Riyadh, Saudi Arabia. He is the author of several papers in journals and conferences. His research interests include recommendation systems, social media mining, big data, and ambient intelligence environment.

ABDULMOTALEB EL SADDIK (Fellow, IEEE) is currently a Distinguished University Professor and a University Research Chair with the School of Electrical Engineering and Computer Science, University of Ottawa. His research focus is on the establishment of digital twins to facilitate the wellbeing of citizens using AI, IoT, AR/VR, and 5G to allow people to interact in real time with one another as well as with their smart digital representations. He has coauthored ten books and more than 550 publications and chaired more than 50 conferences and workshops. He has received research grants and contracts totaling more than $20 M. He has supervised more than 120 researchers and has received several international awards, for example, an ACM Distinguished Scientist, a Fellow of the Engineering Institute of Canada, a Fellow of the Canadian Academy of Engineers and a Fellow of IEEE I&M Technical Achievement Award, IEEE Canada C.C. Gotlieb (Computer) Medal, and the A.G.L. McNaughton Gold Medal for important contributions to the field of computer engineering and science.