applied sciences

MDPI

Article

A Systematic Approach to Healthcare Knowledge Management Systems in the Era of Big Data and Artificial Intelligence

Anh-Cang Phan [1,*], Thuong-Cang Phan [2] and Thanh-Ngoan Trieu [2,3]

[1] Faculty of Information Technology, Vinh Long University of Technology Education, Vinh Long 85110, Vietnam
[2] College of Information and Communication Technology, Can Tho University, Can Tho 94100, Vietnam; ptcang@cit.ctu.edu.vn (T.-C.P.); ngoan.trieuthanh@etudiant.univ-brest.fr (T.-N.T.)
[3] La Faculté Sciences et Techniques, Université de Bretagne Occidentale, 29200 Brest, France
* Correspondence: cangpa@vlute.edu.vn; Tel.: +84-918-204-917

Abstract: Big data in healthcare contain a huge amount of tacit knowledge that brings great value to healthcare activities such as diagnosis, decision support, and treatment. However, effectively exploring and exploiting knowledge on such big data sources exposes many challenges for both managers and technologists. In this study, we therefore propose a healthcare knowledge management system that ensures the systematic knowledge development process on various data in hospitals. It leverages big data technologies to capture, organize, transfer, and manage large volumes of medical knowledge, which cannot be handled with traditional data-processing technologies. In addition, machine-learning algorithms are used to derive knowledge at a higher level in supporting diagnosis and treatment. The orchestration of a knowledge system, big data, and artificial intelligence brings many advances to healthcare. Our research results show that the system fully ensures the knowledge development process, serving for knowledge exploration and exploitation to improve decision-making in healthcare. The knowledge system is illustrated for the detection and classification of high blood pressure and brain hemorrhage in text and CT/MRI image formats, respectively, from medical records of hospitals. It can support doctors to accurately diagnose the diseases to give appropriate treatment regimens.

Keywords: KMS; big data; machine learning; high blood pressure; brain hemorrhage; Spark

check for updates

Citation: Phan, A.-C.; Phan, T.-C.; Trieu, T.-N. A Systematic Approach to Healthcare Knowledge Management Systems in the Era of Big Data and Artificial Intelligence. *Appl. Sci.* **2022**, *12*, 4455. https://doi.org/10.3390/app12094455

Academic Editors: Xiao Chen, ASM Shihavuddin and Dan Zheng

Received: 26 March 2022
Accepted: 25 April 2022
Published: 28 April 2022

1. Introduction

Knowledge represents an important resource that needs effective management to capture, organize, transfer, and apply this kind of intellectual property. A knowledge management system (KMS) is a class of information systems for managing organizational knowledge. Unlike traditional information systems that only focus on capturing, organizing, and managing explicit knowledge, KMS explores and exploits explicit and tacit knowledge. The advancement of knowledge management systems has changed the way organizations operate, especially medical organizations, in which healthcare is a knowledge-intensive industry. Healthcare data come from many sources such as hospital databases, national databases, or private analytic databases. An example of private analytic databases is the Premier Hospital Database, which comprises data from more than 1 billion patient encounters from over 700 private and academic hospitals in the United States, corresponding to approximately 20% of all hospitalizations in the country [1]. Many studies [2–4] leverage the available databases to reveal valuable knowledge, which is meaningful in public healthcare. The large-volume databases including patient information, disease diagnosis, and medical treatment allow for the investigation of rare diseases and uncommon complications that are not always possible with prospective clinical studies. However, the rapid increase in healthcare records in these databases poses many challenges for KMS to improve the decision-making support process. Specially, with the advent of technology in the field of the Internet of Things, many wearable sensor devices are launched to remotely monitor patients' health. This will rapidly enlarge the size of the health records in healthcare systems. The large amount of data needs to be managed and analyzed appropriately. Big data in healthcare contain explicit and tacit knowledge that supports a wide range of medical functions such as disease monitoring, clinical decision support, and healthcare management. Thus, it is necessary to build an effective KMS managing the precious knowledge to support medical diagnosis decision-making in the context of big data and artificial intelligence.

Alavi and Leidner presented discussions about knowledge, knowledge management, and knowledge management systems [5]. They described issues, challenges, and benefits of knowledge management systems [6]. Brent Gallupe considered three levels of knowledge management technologies: tools, generators, and specific KMSs [7]. Some studies discussed knowledge management in the age of big data related to some aspects such as knowledge bases, knowledge discovery, and knowledge fusion. Suchanek and Weikum gave an overview of the methods for building large knowledge bases [8]. Begoli and Horey presented three system design principles that can be integrated into knowledge discovery infrastructure and provided development experiences with big data problems [9]. Dong et al. introduced a web-scale probabilistic knowledge base that employed supervised machine-learning methods in knowledge fusion from existing repositories [10]. These studies considered the presentation of big data in their systems, but they did not provide a comprehensive process of knowledge development. Tretiakov et al. [11] adapted and extended a generic model of knowledge management systems including relevant factors to healthcare. Experiments were conducted on data collected from 263 doctors within two district health boards in New Zealand. Maramba et al. [12] presented a comprehensive synopsis of the challenges in the implementation of computer-based KMS in healthcare institutions. Manogaran et al. [13] proposed a big-data-based KMS supporting clinical decisions. They provided an overview of big data tools and technologies that can be used in KMS. These observed studies remain at the level of knowledge exploration that do not apply new knowledge in concrete practice. Recently, Le Dinh et al. proposed an architecture for implementing big-data-driven knowledge management systems [14]. A knowledge management system in a big data context must fully ensure the development process of knowledge including four stages: capture, organize, transfer, and apply. The study stays on the abstract level of KMS without any implementation.

In order to overcome the above challenges, we propose to build a big-data-driven healthcare knowledge management system supporting the diagnostic decision in a parallel and distributed environment. The large-scale healthcare system ensures a complete and comprehensive knowledge development process, including knowledge exploration and knowledge exploitation. Additionally, the involvement of artificial intelligent and big data processing is to provide real-time diagnosis decision supports with the massive volumes of medical records for a reasonable response time. The proposed healthcare knowledge management system for supporting medical diagnosis includes four layers: a data layer, an information layer, a knowledge layer, and an application layer. An illustration of the proposed system is presented using machine-learning techniques in the knowledge layer to generate knowledge for hypertension and brain hemorrhage diagnosis. Data used in this system are collected from several hospitals and health-monitoring devices. Hypertension is one of the most leading causes of disability and death worldwide. According to the World Health Organization (WHO), an estimated 9.4 million deaths are caused by high blood pressure. This dangerous disease needs to be promptly detected and treated to limit the risks of death as well as disease complications. We use decision trees to generate knowledge for hypertension diagnosis and classification. Decision trees learn and generate simple rules from a complex decision-making process that is similar to the way of human thinking. In addition, we use deep-learning techniques to generate knowledge for brain hemorrhage detection and classification. A brain hemorrhage is a type of stroke that is caused by an artery bursting in the brain. Stroke is the second leading cause of death according to the World Health Organization. The diagnosis of the disease is based on cerebral CT/MRI images; thus, we proposed to use deep-learning techniques for hemorrhage detection and classification. The trained model with Faster R-CNN Inception ResNet v2 achieves the mean average precision of 79% in classifying four types of brain hemorrhage.

The structure of this paper is organized as follows. Section 2 presents the background related to knowledge management systems in a big data context. The proposed method in Section 3 provides in details our proposed architecture to build a knowledge management system supporting medical diagnosis decisions. The next section introduces a healthcare knowledge management system with the proposed architecture for high blood pressure and brain hemorrhage diagnosis. The conclusion of the paper is presented in the last section.

2. Background

2.1. Knowledge Management Systems

Knowledge management systems have a dramatic impact on the decision-making support of organizations. However, an effective KMS needs to ensure the whole process of knowledge management, including knowledge exploration and knowledge exploitation. Le Dinh et al. proposed an architecture for big-data-driven knowledge management systems including a set of constructs, a model, and a method [14]. This architecture has complied with the requirements of the knowledge development process and the knowledge management process. The architecture is presented in Figure 1, consisting of four layers: data layer, information layer, knowledge layer, and process layer.

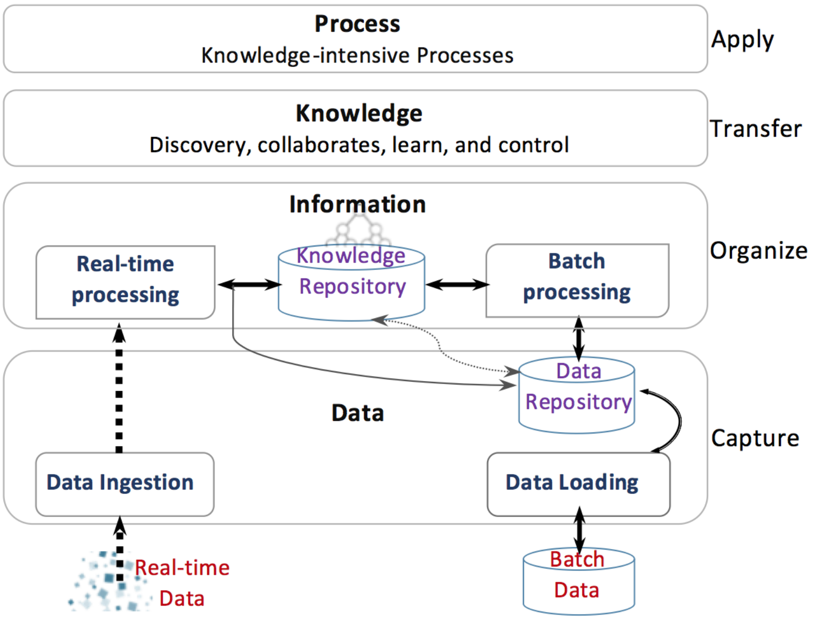


Figure 1. An architecture for big-data-driven knowledge management systems.

Constructs are knowledge objects, which are classified based on their level of development. Knowledge is defined as information about facts, concepts, ideas, and judgments obtained through experience [5]. It is classified as data, information, knowledge, and wisdom through a knowledge development process [15]. The term wisdom was changed to understanding by Le Dinh et al. [14] in their proposed architecture for knowledge-based decision support systems instead of the best decision-making support systems.

A method is the activities that generate knowledge objects in the development process. Four main activities correspond to four levels of knowledge objects in the knowledge management process (Figure 2), which are capture, organization, transfer, and application [16]. Data are captured and stored in a knowledge repository. Data are organized to become useful information. Information is transferred to become knowledge, and this knowledge is applied to synthesize new knowledge from existing knowledge for better understanding.

These activities ensure knowledge management systems can explore and exploit different kinds of knowledge. Knowledge exploration is a process that concerns capturing and organizing knowledge, whereas knowledge exploitation concerns the transformation and application of knowledge.

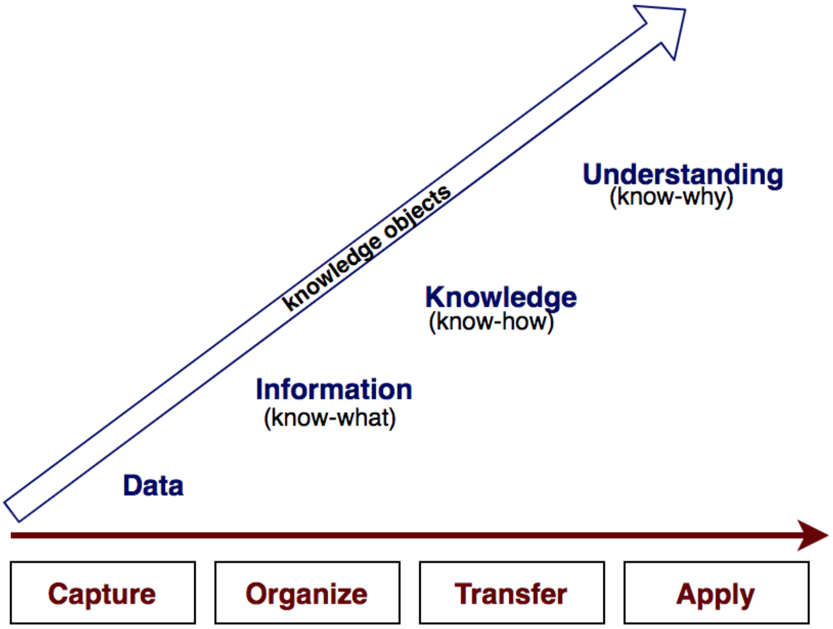


Figure 2. Knowledge development process.

A model is a relationship between knowledge components and knowledge objects. A knowledge object is considered as a collection of knowing, namely a knowledge component. Knowledge components can be classified as structure, transition, and coherence of knowledge [17]. The structure of knowledge is represented by a know-what component, which is the information that can answer simple questions related to a phenomenon such as what, who, when, and where. The transition of knowledge is represented by a know-how component, which is the appropriate knowledge about the process of the phenomenon. The coherence of knowledge is represented by a know-why component, which provides understanding about the principles of the phenomenon.

2.2. Apache Spark and Apache Kafka

Apache Spark is an open-source computing framework, originally developed at the University of California Berkeley in 2009 [18]. Apache Spark consists of 5 main components: Spark Core, Spark Streaming, Spark SQL, MLlib, and GraphX [19]. Spark Core is the main component, which is the basic general execution engine to build other functions on it. Spark Core supports multiple application programming interfaces with languages such as Java, Scala, and Python.

Spark's processing speed is achieved with the capability of in-memory computing and parallel and distributed computing on a cluster of computers. Given a task, Spark allows the division of this task into more manageable chunks. Spark will then run these small tasks in memory on clusters of many different computing nodes. Spark consists of a master node and multiple worker nodes. Spark Driver will contact the master node to coordinate the workers, where there are executors for executing the tasks (Figure 3).

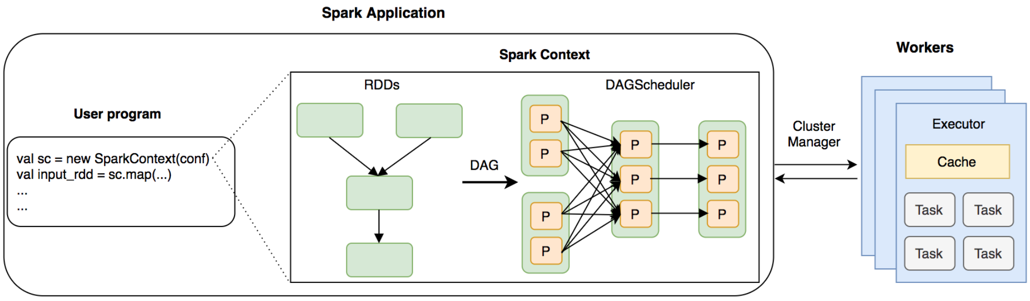


Figure 3. Deploying computation in Spark.

Spark allows real-time data processing, performing processing right on the received data with Spark Streaming. It enables powerful analytics on both real-time and historical data. Spark Streaming easily integrates with many popular data sources, including HDFS, Flume, Kafka, and Twitter. One thing that should be noted is that Spark does not have its own distributed file system; thus, users can use other file systems such as HDFS, HBase, and Cassandra.

Apache Kafka is a high-performance streaming message platform that was first introduced in 2011 for collecting and delivering high volumes of data [20]. Kafka is distributed and scalable providing an API similar to a message passing system to process a huge amount of data in real time. There are some basic concepts used in the overall architecture of Kafka (Figure 4). Topic is a category name to which messages are stored and published. A topic is divided into multiple partitions in which data are stored in an immutable order and assigned an ID called offset. A producer can publish messages to a topic. The published messages from the producers are stored on a set of servers called brokers. A broker allows consumers to fetch messages by topic, partition, and offset. A consumer consumes messages of the subscribed topics by pulling data from the brokers. Exactly one consumer in a consumer group consumes each partition of a topic.

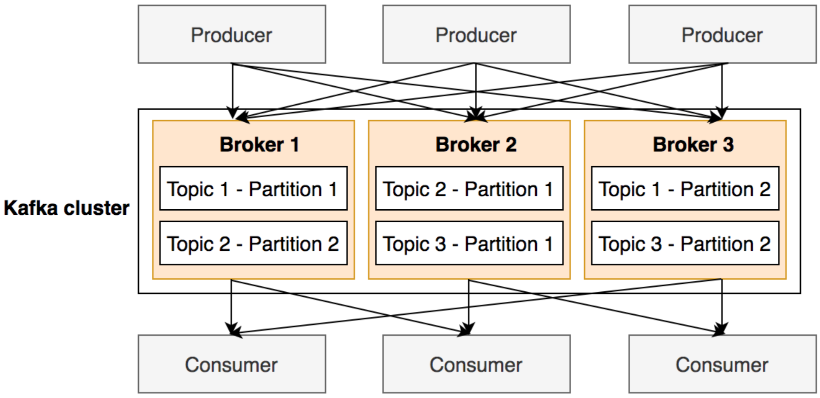


Figure 4. Kafka architecture.

2.3. *Artificial Intelligence*

Machine learning is a subset of artificial intelligence that allows a machine to automatically learn from historical data to simulate human behavior. In this study, we use two machine-learning techniques: decision trees and deep learning.

2.3.1. Decision Trees

A decision tree is a model of supervised machine-learning algorithms. It comprises a hierarchical tree structure used to classify objects based on sequences of rules [21], and it can be applied to the problems of classification and regression. The decision tree is a popular method of data mining using multi-criteria decision analysis. Moreover, a decision tree is a tree where each path from the root node to the leaf is a rule. Each node represents an attribute, each branch represents a test value of the corresponding attribute, and each leaf is a label defined by the rule. An example of a decision tree is shown in Figure 5. Outlook is the root node, and the leaves are labeled for predictions.

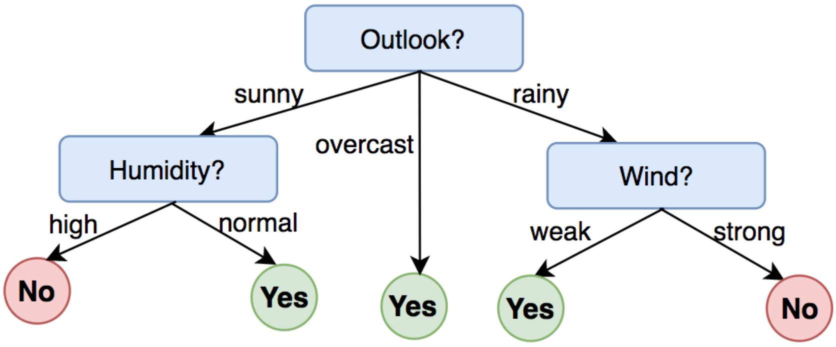


Figure 5. Example of decision tree.

2.3.2. Deep-Learning Neural Networks

Faster R-CNN for Prediction

The Faster Region-based Convolutional Neural Network (Faster R-CNN) [22] is a deep convolutional network used for object detection. The network can accurately and quickly predict the locations and classification of different objects. It is an improvement of the Fast R-CNN model [23] that uses the region proposal network (RPN) instead of the selective search algorithm. Faster R-CNN includes two main phases, which are a phase using RPN to generate proposed regions and a phase segmenting and classifying objects in the proposed regions. An input of any size is accepted by RPN, and it then outputs the proposed regions with a probability of containing objects.

Classification Loss and Localization Loss determine the Loss function of Faster R-CNN as shown in Equations (1) and (2).

$$Loss(\{p_i\}, \{t_i\}) = \frac{1}{N_{cls}} \Sigma_i L_{cls}(p_i, p_i^*) + \lambda \frac{1}{N_{reg}} \Sigma_i p_i^* L_{reg}(t_i, t_i^*) \tag{1}$$

$$smoothL1(x, y) = \begin{cases} 0.5(x_i - y_i)^2 & if \mid x_i - y_i \mid < 1 \\ \mid x_i - y_i \mid -0.5 & otherwise \end{cases} \tag{2}$$

where i is the index of the anchor in mini-batch; p_i is the predicted probability of anchor i being an object; the ground-truth label value p_i^* is 1 if the anchor is positive, and 0 otherwise; t_i is a 4-dimensional vector represents the coordinate values of the predicted bounding box; t_i^* is a 4-dimensional vector represents the coordinate values of the ground-truth box corresponding to the positive anchor; L_{cls} is the log loss of 2 classes (object and non-object); and L_{reg} is the $smoothL1$.

Inception ResNet v2 for Feature Extraction

Inception ResNet v2 [24] is a convolutional neural network that is trained on more than a million images from the ImageNet database. It is a combination of Inception and Residual network architectures. Inception is an artificial neural network for feature extraction with a low error rate. ResNet is a deep network with hundreds or even thousands of layers using a skip connection technique. In this study, Inception ResNet v2 is used as a pretrained CNN for a backbone in Faster R-CNN to extract features due to computational time limitations.

2.3.3. Evaluation Metrics

We analyze the efficiency of classification algorithms based on precision and recall. Precision is the ratio of the correct positive predictions to the total number of positive predictions. The higher the precision, the better the model is on positive classification.

Recall is the ratio of the correct positive predictions to all samples belonging to the positive group. The higher the recall, the lower the number of missed positive cases. Precision (P) and recall (R) are calculated as in Equation (3).

$$P = \frac{TP}{TP + FP} \qquad R = \frac{TP}{TP + FN} \tag{3}$$

F1-score is the weighted average of precision and recall, and is defined as the harmonic mean function between precision and recall (Equation (4)).

$$F(\beta) = (1 + \beta^2)(\frac{P * R}{\beta^2 * P + R}) = 2\frac{P * R}{P + R} \tag{4}$$

A mean average precision (mAP) is another popular method to evaluate the model accuracy. It is calculated as in Equation (5) where AP_i is the average precision for ith class in N classes.

$$mAP = \frac{1}{N}\sum_{1}^{N} AP_i \tag{5}$$

3. Proposed Method

The problem posed in this study is to build a knowledge management system to support medical diagnosis decisions in a big data environment. It must provide a complete knowledge development process, including knowledge exploration and knowledge exploitation. Based on the research of Le Dinh et al., we have proposed an architecture for a knowledge management system supporting medical diagnosis including four layers: data layer, information layer, knowledge layer, and application layer (Figure 6). This knowledge management system ensures all four stages of the knowledge development process, including data, information, knowledge, and understanding, corresponding to four main activities, which are capture, organize, transfer, and apply. The objective of this study is to present the architecture for medical diagnosis decision-supporting systems by collecting and analyzing big data. This proposal addresses two major challenges: knowledge management and knowledge organization from disparate data sources.

The system processes two types of data: batch data (patient records collected over a long time period) and real-time data (collected from wearable devices). The batch data are loaded into the data lake (HDFS) and the real-time data are ingested into the processing system with Kafka and Spark streaming. With a large amount of medical data, the system will filter out useful information for disease diagnosis and classification, preprocess information, and store information into HBase. The information will be used for knowledge transformation to create machine-learning models. New knowledge is created and made available to users through queries from websites or wearable devices.

3.1. Data Layer

There are two data sources used in this study, including historical datasets collected from hospitals and real-time data collected from patients via health-monitoring wearable devices. The batch data are loaded into Hadoop Distributed File System (HDFS), a well-known fault-tolerant distributed file system. HDFS is designed to store very large datasets reliably and to stream those datasets at high bandwidth to user applications. The real-time data are ingested into the system with Apache Kafka, a distributed, reliable, high-throughput and low-latency publish–subscribe messaging system. Kafka has become popular when it and Apache Spark are coordinated to process stream data as well as to use both of their advantages. We use Kafka to ingest real-time event data, streaming it to Spark Streaming. The data can be in text format or images, especially CT/MRI images that are commonly used in medical diagnosis. These raw data are collected and fed into the system for storage at the data layer.

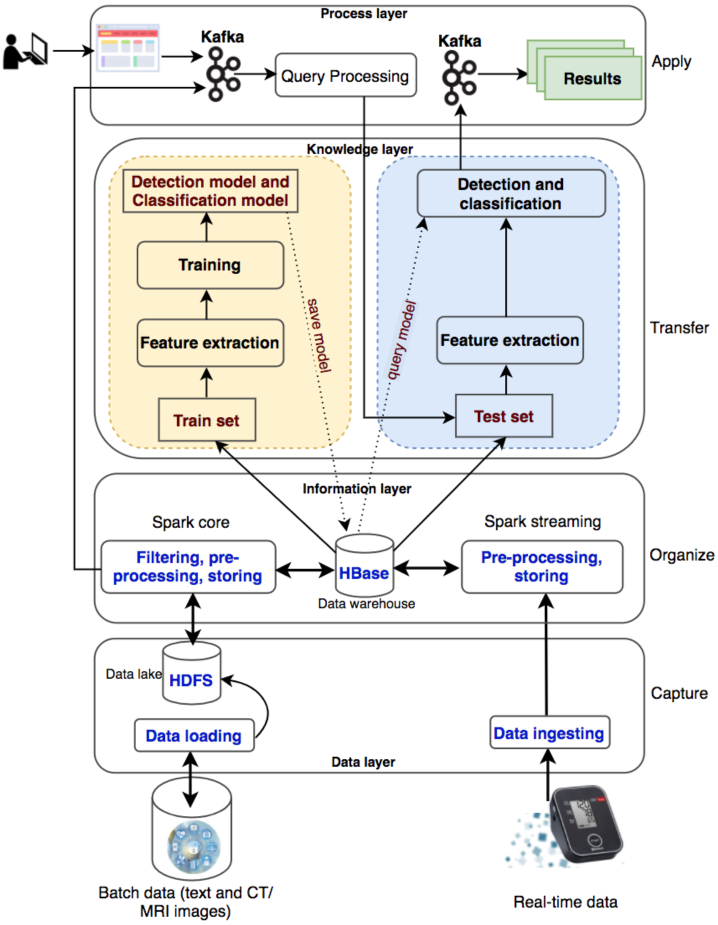


Figure 6. Proposed architecture for healthcare knowledge management systems.

3.2. Information Layer

Data will be sorted, organized, and filtered accordingly to transform into meaningful information in an organized and retrievable form. This information will be stored as datasets on a distributed file system HBase to serve for distributed and parallel processing in a big data environment. Apache HBase is a distributed column-oriented NoSQL database built on top of HDFS. The system requires the ability to handle batch and real-time data. Consequently, we use Apache Spark for both the batch and real-time data processing. Spark has emerged as the next-generation big-data-processing engine because it works with data in memory that are faster and better able to support a variety of compute-intensive tasks. Spark Core processes the batch data from HDFS to organize content according to their semantics and to create and maintain the knowledge base (HBase) as an organizational memory. Spark Streaming involves mapping continual input of the data from Kafka into real-time knowledge views. Every single event is sent as a message from Kafka to the Spark Streaming. Spark Streaming produces a stream and executes window-based operations on them.

The data collected from the hospital management system consist of many tables and many data fields. Depending on the goals of the medical diagnostic support systems, the appropriate data should be extracted. The historical datasets collected from hospitals will be used for the knowledge generation process, which is the input to the knowledge layer. These data are authentic, and the diagnostic results are given by the doctors with high professional confidence to help the labeling process in building knowledge models more effectively.

3.3. Knowledge Layer

Machine-learning algorithms can be used in the Spark distributed environment to build models for knowledge generation consisting of two phases: the training phase and the testing phase. Spark MLib is a core component to execute the learning service that allows for quickly experimenting and building data models. The appropriate models supporting diagnosis decisions will be made based on accuracy. In this layer, it is necessary to perform preprocessing of the data, which is to select the necessary information for the construction of a diagnosis support system. The diagnosis results previously given by doctors are used for labeling purposes. After data preprocessing, 70% of the random dataset will be used for the training phase and 30% for the testing phase.

The machine-learning algorithms used in the knowledge layer are decision trees and deep neural networks. Decision trees have been successfully used in a wide range of fields such as speech recognition, remote sensing, and medical diagnosis. The reason for choosing a decision tree at the knowledge layer here is that the patient records for hypertension are all in text format. The decision tree uses input data to learn and generate knowledge with the same rules as to how humans think. It breaks down a complex decision-making process into simple rules that are simple to understand and suitable to use for datasets of diverse attributes and data types. Deep learning with Faster R-CNN Inception ResNet v2 is another machine-learning algorithm to be used in the knowledge layer for brain hemorrhage diagnosis. Deep-learning techniques have been successfully applied in a wide range of fields, especially in medical images analysis.

3.3.1. Training Phase

In this phase, we perform feature extraction on the input dataset and then train machine-learning models. Model training is performed in a distributed environment and stores the trained model on distributed file systems (Figure 7). We build machine-learning models with the extracted feature dataset.

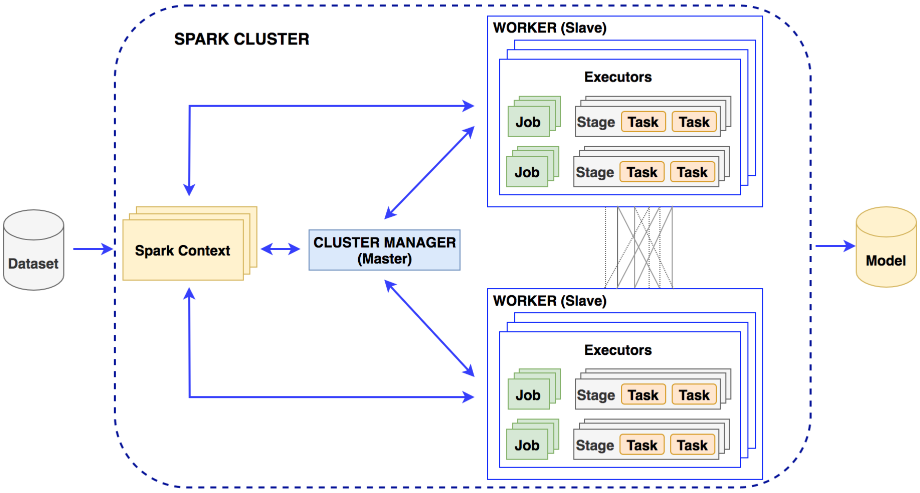


Figure 7. Training phase in a Spark cluster.

3.3.2. Testing Phase

We extract features for the testing set, thereby evaluating the accuracy of the trained models with the test set. The trained model is used to predict whether or not a patient has a disease. The execution of queries in this phase is also implemented in a distributed parallel environment. Machine-learning models are used in the testing phase to evaluate the accuracy of the predictions. The models' performance can be evaluated with precision, recall, and F1 score. The appropriate models for the problem will be stored on a distributed storage system for future use.

3.4. Process Layer

In this layer, the applications are built to input patient information into the system and give outputs about diagnosis and diseases classification. The applications are designed to perform patient data entry and then execute knowledge queries to return new knowledge about the patient's health status. The execution of queries in this layer is implemented in a distributed environment.

4. Healthcare Knowledge Management Systems

We illustrate our proposed approach in building healthcare knowledge management systems for hypertension and brain hemorrhage diagnosis. The input dataset is collected from several hospitals in Mekong Delta and stored in the Postgres database. We only use some main data tables that contain the data needed for the application related to high blood pressure and brain hemorrhage. These data are the patient records in text format and CT/MRI images. The text data fields include 168,793 data records with 13 fields (gender, age, height, weight, temperature, systolic blood pressure, diastolic blood pressure, pulse, respiratory rate, head circumference, chest circumference, symptoms, and diagnosis results). In addition to the text data, 479 patient records contain a cerebral CT/MRI image size 512 × 512. The image dataset includes 79 images of epidural hematoma (EDH), 54 images of subdural hematoma (SDH), 90 images of subarachnoid hemorrhage (SAH), and 256 images of intracerebral hemorrhage (ICH). The systems are built on a Spark cluster of three nodes (one master and two slaves) with the configuration of the master node being Intel Core i7 3.2 Ghz 4 CPUs 16GB RAM with Nvidia Tesla P100 GPU and the slave node being Intel Core i7 3.2 Ghz 1 CPU 4GB RAM. The operating system is Ubuntu 20.04.1 LTS 64 bit, and the versions of installed software are Java 1.8, Hadoop 3.2.1, and Spark 3.0.1. The library used to support the training of the network models is TensorFlow GPU version 1.5. Models are trained on Spark cluster to be able to execute on multiple computing nodes to shorten training time. When configuring the Spark cluster, we can change the variables of the cluster such as the driver memory, the number of executor cores, and the executors' memory (—executor-memory 2 g—driver-memory 2 g—total-executor-cores 5).

4.1. High Blood Pressure Diagnosis Support

Blood pressure is the blood force exerted against vessel walls as it moves through the vessels [25]. Blood pressure is expressed as two numbers: systolic pressure and diastolic pressure. Systolic is the higher number, which corresponds to the period when the heart beats to push the blood in the arteries. Diastolic is the lower number, which corresponds to the rest period between two consecutive heartbeats. Typically, high blood pressure is when the blood pressure measured in medical facilities is greater than or equal to 140/90 mmHg. According to the seventh report of the Joint National Committee on Prevention, Detection, Evaluation, and Treatment of High Blood Pressure (JNC 7) [26], the classification of blood pressure for adults aged 18 and older is presented in Table 1.

Table 1. Classification of blood pressure for adults.

Class	**Systolic**	**Diastolic**
Normal	<120	and <80
Prehypertension	120–139	or 80–89
Stage 1 hypertension	140–159	or 90–99
Stage 2 hypertension	≥160	or ≥100

In this study, we build a big-data-driven healthcare knowledge management system supporting high blood pressure diagnosis and classification. Besides the previous diagnosis results from doctors, we rely on Table 1 to be able to label the levels of high blood pressure disease based on systolic and diastolic blood pressure. The machine-learning algorithm to be used in the knowledge layer for this type of disease is decision trees.

Feature extraction: We choose VectorAssembler1 to transform the information in the dataset into feature vectors. The string data in the input dataset will be converted into input requirements of VectorAssembler. This conversion is performed by indexing string data with StringIndexer2 and representing the indexes as binary vectors through OneHotEncoder3. We use VectorAssembler on the input dataset and generate a feature extraction model. This feature extraction model will be stored in the database for reuse in the testing phase.

Model training: We build decision trees with the extracted feature dataset. One of the important variables to keep in mind when building a decision tree is maxDepth. It allows us to decide the depth of the tree, and we can tweak this parameter to improve the accuracy of the model. However, the deeper the tree, the higher the likelihood of over-fitting problem occurring. Over-fitting occurs when the models make rules that fit exactly against the training data but will make wrong predictions on testing data.

4.1.1. Decision Tree for High Blood Pressure Detection

Training Phase

Preprocessing: The text data have a lot of empty data, zero value data, and even non-viable values that will affect the operations of the knowledge layer. Therefore, data preprocessing will remove non-viable values from the dataset. The solution to empty data fields is filling values using mathematical interpolation. This dataset is saved as a csv extension file and put on HBase for later use in distributed environments.

We label the data records based on the diagnosis results, which are concluded by professional doctors with high reliability. The data record is labeled 1 if the patient is diagnosed with high blood pressure and 0 otherwise. After labeling, we process the string information in the dataset to build a feature extraction model and receive the feature vectors.

Model training: We fit a decision tree with a ratio of 70/30 for training and testing phases. A classification decision tree is built with the train set, and then we will use the test set to evaluate the model performance. Table 2 contains the information of the dataset after labeling and feature extraction. This information is obtained during the steps we take before dividing train/test sets.

Table 2. Examples of data before training models.

Symptoms	Diagnosis	Label	Index	Symptoms Classification	Features
Headache, vomit	Intracranial injury	0	194	(25,152, [194], [1.0])	(25,163, [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 205], [17.0, 100.0, 60.0, 80.0, 18.0, 1.57, 22, 53, 48.0, 37.0, 1.0])
Fiver	Chickenpox	0	7	(25,152, [7], [1.0])	(25,163, [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 18], [1.0, 36.0, 140.0, 60.0, 78.0, 20.0, 1.7, 39, 68, 50.0, 39.0, 1.0])
Tired	Hypertension	1	1	(25,152, [1], [1.0])	(25,163, [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 12], [49.0, 210.0, 140.0, 104.0, 22.0, 1.73, 40, 55, 80.0, 37.0, 1.0])
Abdominal pain	Acute appendicitis	0	0	(25,152, [0], [1.0])	(25,163, [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11], [23.0, 110.0, 70.0, 87.0, 20.0, 1.46, 40.0, 50.0, 40.0, 37.0, 1.0])
Dizzy	Vestibular dysfunction; Hypertension	1	4	(25,152, [4], [1.0])	(25,163, [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 15], [1.0, 53.0, 170.0, 100.0, 84.0, 18.0, 1.5, 42, 55, 50.0, 37.0, 1.0])

In addition, based on the trained model, we use the featureImportances function supported by PySpark library to select variables that have an important influence on the disease diagnosis in the dataset. The importance of a variable is weighted by Gini-importance defined by the total decrease in node impurity. It is calculated by the number of samples that reach the node, divided by the total number of samples. The higher the value, the more important the feature is. We can rely on this result to remove unimportant data fields to reduce training time as well as increase the accuracy of the model. The results we obtained from the featureImportances are shown in Figure 8.

We decided to remove two unimportant data fields (head circumference and chest circumference) and retrain the models with the dataset consisting of only 11 data fields. We train different decision tree models by varying the tree depth as well as performing the training phase in a distributed environment with three proposed scenarios.

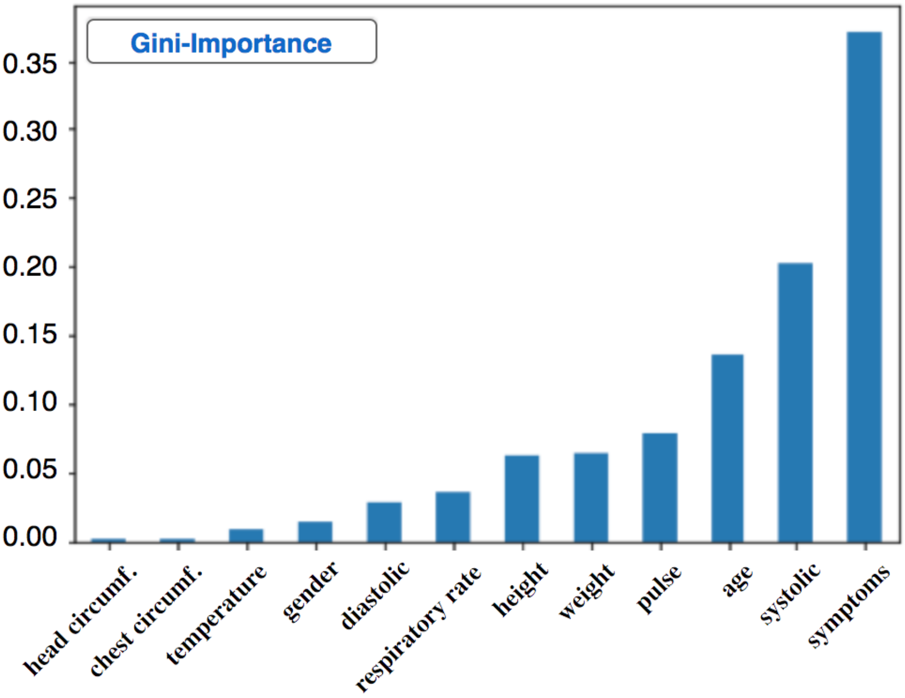


Figure 8. Feature importance in predicting high blood pressure.

Training results: We construct decision trees with different depths. Each tree will have rules that give different prediction results. A tree of depth n will inherit inner branches from a tree of depth $n-1$ and has additional conditions for making predictions. An example to illustrate a decision tree with a depth of 4 is shown in Figure 9.

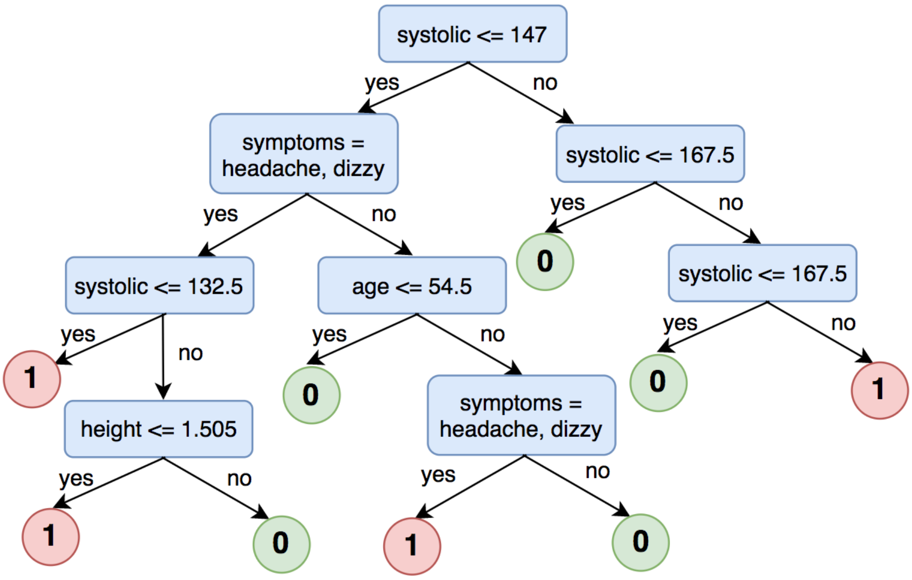
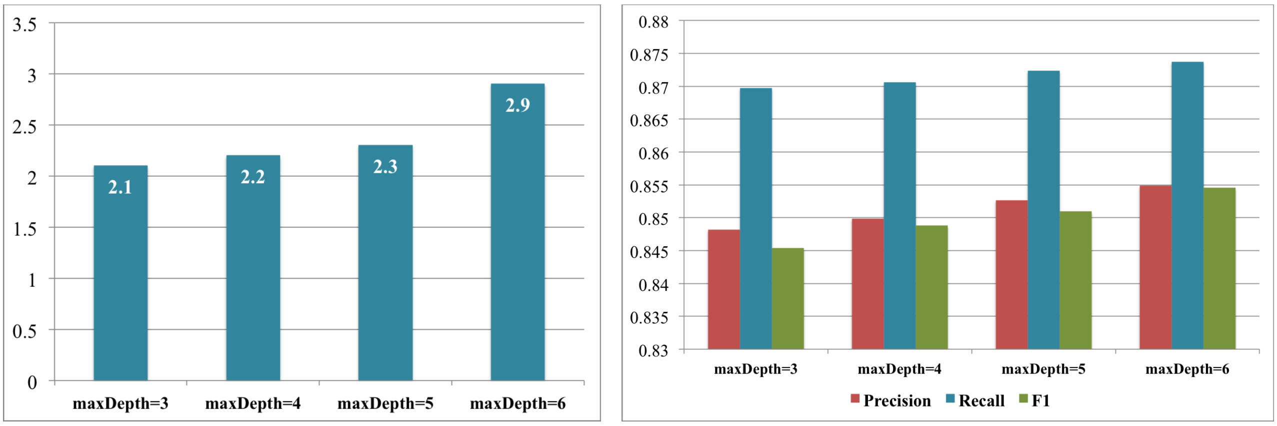


Figure 9. Decision tree of depth 4 for the problem high blood pressure detection.

In addition, based on the decision tree models and the rules generated, we found that several health factors of the patient are closely related to high blood pressure. For example, a patient with the systolic blood pressure of over 147 usually has some symptoms such as headache, dizziness, and fatigue. People over the age of 55 are likely to have a high risk of hypertension. We train the models on a Spark cluster, and the training time is presented in Figure 10a. The deeper the tree, the more time it spends on the training process.

Testing Phase

After finishing the training process, we evaluate the detection models by applying the models for high blood pressure detection on the testing set. The accuracy of the models received is presented in Figure 10b. The precision of the models with different tree depth levels reaches 84% to 87%. After the process of training and evaluating the results of the models, we choose to stop training at a tree depth of 6 because the generated rules are consistent with reality. These things considered, if we increase the depth of the tree, we find that redundant branches start to appear, and the decision trees fall into over-fitting.



(**a**) (**b**)

Figure 10. Training time and accuracy of the detection models. (**a**) Training time; (**b**) Accuracy.

4.1.2. Decision Tree for High Blood Pressure Classification

Model training: The classification of high blood pressure is based on Table 1. We perform labeling by comparing the patient's systolic and diastolic blood pressure to make the classification as follows.

- Label 0: systolic < 120 and diastolic < 80
- Label 1: systolic ≥ 120 and diastolic ≥ 80
- Label 2: systolic ≥ 140 and diastolic ≥ 90
- Label 3: systolic ≥ 160 and diastolic ≥ 100

The classification of the disease is conducted after the disease detection; thus, we do not pay attention to label 0. We train decision trees for classification problems on the same dataset with the ratio of 70/30 for train/test sets on the three proposed scenarios.

Results: Similar to the detection of hypertension, we build a classification model of high blood pressure with decision trees at different depths. We choose to stop training at a tree depth of 4 because as the depth increases, redundant branches start to appear, and the tree falls into over-fitting. An example of a decision tree that classifies hypertension with a tree depth of 4 is shown in Figure 11.

The classification models are trained on a Spark cluster. The training time is presented in Figure 12a. The deeper the tree, the more time it spends on the training process. We evaluate the classification models based on precision, recall, and F1-score. The accuracy of the models received is presented in Figure 12b. We receive a precision of over 92% all over the three models.

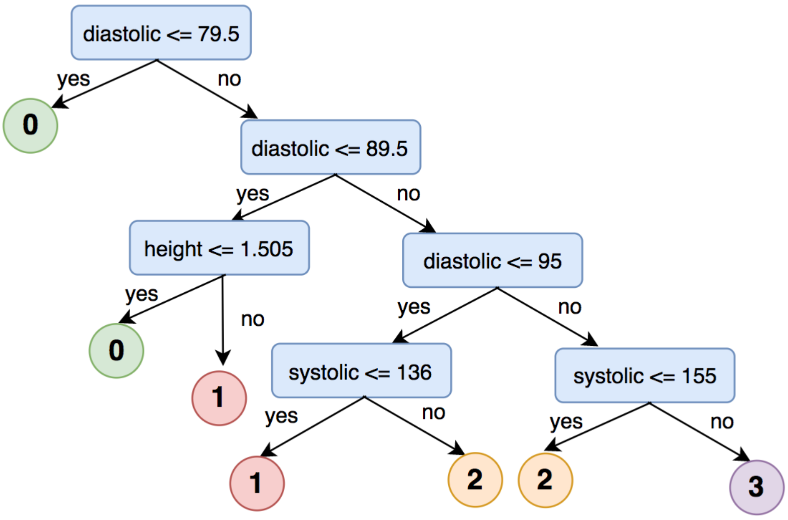
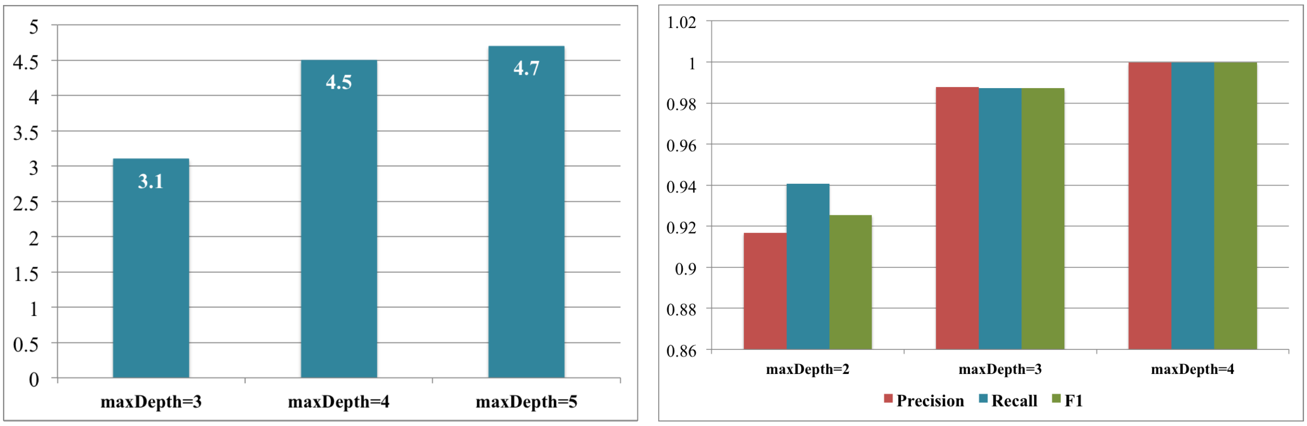


Figure 11. Decision tree of depth 4 for the problem of high blood pressure classification.



(**a**) (**b**)

Figure 12. Training time and accuracy of the classification models. (**a**) Training time; (**b**) Accuracy.

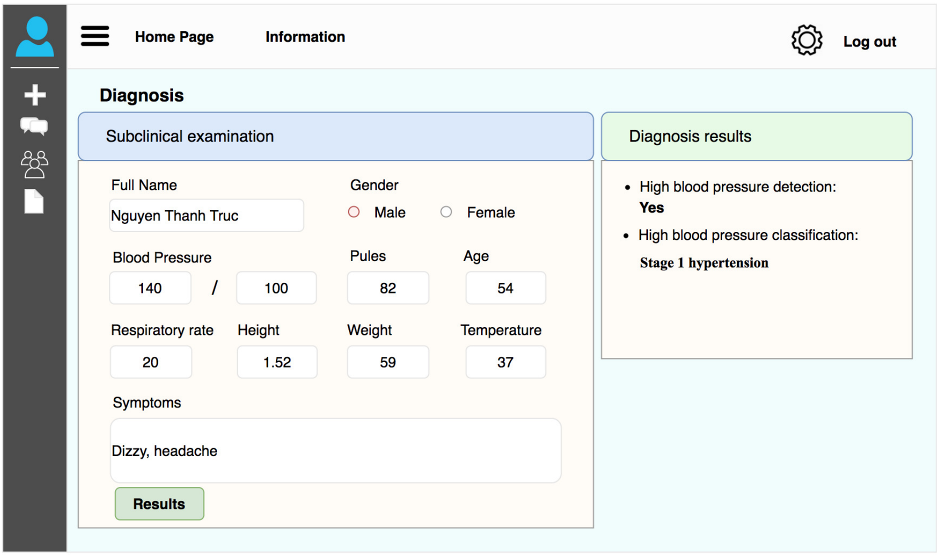
4.1.3. Application

The application is designed for users to enter the medical information needed for the diagnosis and classification of high blood pressure. The system will generate new knowledge of disease diagnosis and classification based on user-supplied information and previously learned knowledge. The application user interface is presented in Figure 13.

The problem here is the conflicts of data that are sent to the system continuously. Therefore, we chose to integrate Kafka into the system to control the continuous querying process. Kafka has the ability to transmit large amounts of data in real time; even if the data have not been transmitted to the receiver, they are stored in a queue to ensure data security.

4.2. Brain Hemorrhage Diagnosis Support

Brain hemorrhage is a dangerous disease, being a type of stroke that can lead to death or disability. There are four common types of cerebral hemorrhage [27]: epidural hematoma (EDH), subdural hematoma (SDH), subarachnoid hemorrhage (SAH), and intracerebral hemorrhage (ICH). Hypertension is the most common cause of primary intracerebral hemorrhage. To detect the brain hemorrhage, doctors usually rely on the Hounsfield Units (HU) of the hemorrhage region in a CT/MRI image. Thus, we propose a diagnosis supporting system for brain hemorrhage detection and classification using HU values. The machine-learning algorithm to be used in the knowledge layer for this type of disease is deep learning, which is mentioned in this study as Faster R-CNN Inception ResNet v2.



Diagnosis

Subclinical examination

Full Name: Nguyen Thanh Truc
Gender: ○ Male ○ Female
Blood Pressure: 140 / 100
Pules: 82
Age: 54
Respiratory rate: 20
Height: 1.52
Weight: 59
Temperature: 37
Symptoms: Dizzy, headache

Diagnosis results

- High blood pressure detection: **Yes**
- High blood pressure classification: **Stage 1 hypertension**

Figure 13. Application of high blood pressure detection and classification.

Hounsfield unit represents different types of tissue on a scale of −1000 (air) to 1000 (bone). Table 3 illustrates different tissues with their HU density. The hemorrhagic region will have HU values in the range of 40 to 90. The HU values are calculated by Equation (6) with p_{value} being the value of each pixel and r_{slope} and $r_{intercept}$ being the values stored in CT/MRI images.

$$HU = p_{value} * r_{slope} * r_{intercept} \tag{6}$$

Table 3. HU density on CT/MRI images.

Matter	Density (HU)
Air	−1000
Water	0
White matter	20
Gray matter	35–40
Hematoma	**40–90**
Bone	1000

4.2.1. Training Phase

Preprocessing: The CT/MRI images will be converted into digital images (.jpg) according to the HU values. The location of brain hemorrhage is determined by HU values; thus, after preprocessing, we will have a digital images dataset with highlighted hemorrhagic regions. The hemorrhagic regions will be labeled with the supervision of specialists.

Feature extraction: We perform feature extraction using a pretrained CNN of Inception ResNet v2 as the backbone of the Faster R-CNN to reduce the computation time. This step helps to quickly classify brain hemorrhage.

Model training: The extracted features are trained on Faster R-CNN. This training process is monitored with the Loss value. When the Loss value is not improved (or not decreased), we stop the training process. The Loss value of the model is very low (below 10%) after 60,000 training steps, as illustrated in Figure 14. This means that the error rate in the brain hemorrhage prediction of the proposed model is very low.

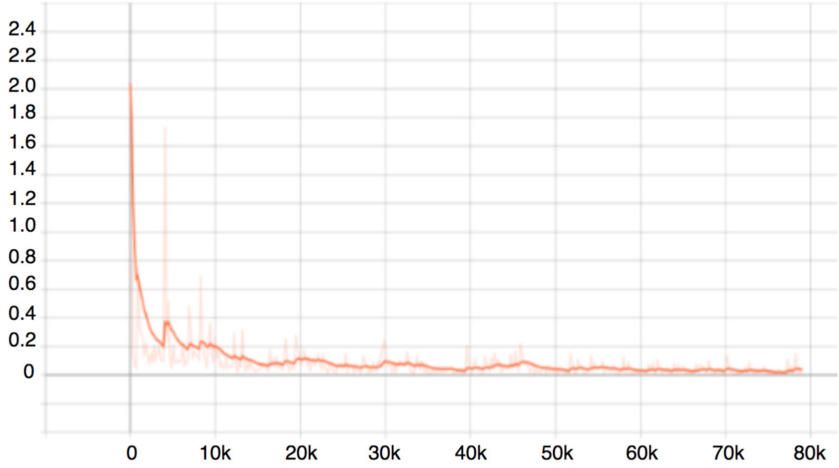


Figure 14. Loss values over training steps.

4.2.2. Testing Phase

After the training process, we evaluate the proposed model for brain hemorrhage detection and classification on the test dataset. The preprocessing and feature extraction are also performed on the testing set before evaluating the model. The trained Faster R-CNN Inception ResNet v2 is then used to detect and classify four common types of brain hemorrhage. It can correctly detect the contours of entire hemorrhage regions with an accuracy of 100%. An example of multiple hemorrhages detection on an image is presented in Figure 15. It can predict bleeding time from 2 to 3 days, recognize hemorrhage type as ICH and SAH, and accurately segment bleeding regions.

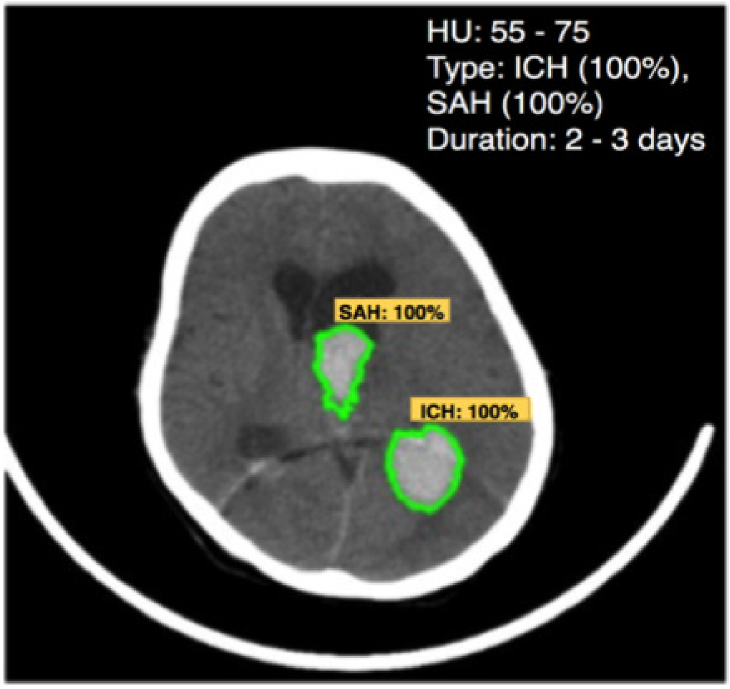


Figure 15. Multi- brain hemorrhages segmentation.

The average precisions (AP) of the proposed model for four types of brain hemorrhage (EDH, SDH, SAH, and ICH) are 0.7, 0.59, 0.72, and 0.71, respectively (Figure 16). This model gives the mAP value of 0.68 for the detection and classification of four classes of brain hemorrhage. The results show that the system can support doctors in accurately diagnosing cerebral hemorrhage and providing appropriate treatment regimens.

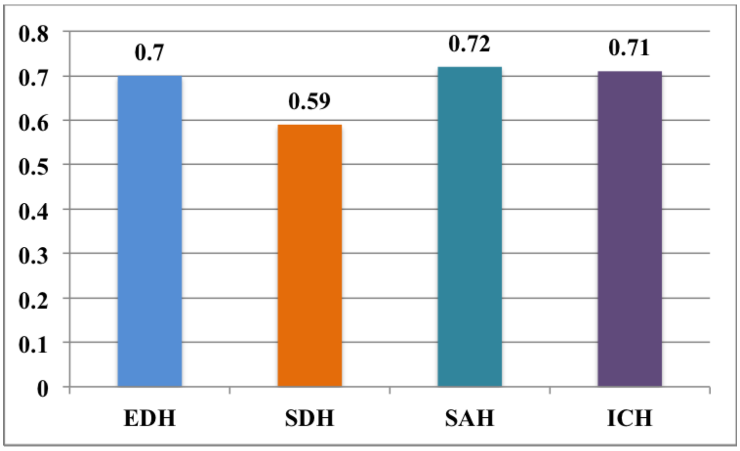


Figure 16. Average precision (AP) of four brain hemorrhage types.

5. Conclusions

Big data give many opportunities along with many challenges in building knowledge management systems, especially in healthcare, a knowledge-intensive industry. In this study, we have proposed a healthcare knowledge management system to improve medical diagnosis decision making in the context of big data and artificial intelligence. It ensures the systematic knowledge development process in which knowledge exploration is performed through machine-learning algorithms and knowledge exploitation is performed through the application of machine-learning models for medical diagnosis. All of these exploration and exploitation activities are conducted in a big data environment. The system is built with the most popular big-data-processing technologies such as Spark, HDFS, HBase, and Kafka.

We illustrate the healthcare knowledge system for the detection and classification of hypertension and brain hemorrhage. The decision tree models are used in the knowledge layer of the system with an accuracy of over 84% for high blood pressure detection and over 92% for high blood pressure classification. In the process of building a decision tree, we rely on the Feature Importance to remove unnecessary data fields in improving model accuracy and optimizing the model training time. The deep neural network with Faster R-CNN Inception ResNet v2 is used in the knowledge layer for brain hemorrhage detection with mAP of 0.68. The data used in the study are collected from several hospitals in the Mekong Delta of Vietnam and health-monitoring devices. The expansion to use other large databases such as the Premier Hospital Database is necessary for this system to be able to take advantage of medical knowledge in different regions. Afterward, we will be able to expand the scope of our research with a wide range of diseases to better serve public healthcare.

Author Contributions: Conceptualization, T.-C.P. and A.-C.P.; methodology, A.-C.P. and T.-C.P.; software, A.-C.P.; validation, T.-N.T. and T.-C.P.; formal analysis, A.-C.P.; investigation, A.-C.P. and T.-C.P.; resources, A.-C.P. and T.-C.P.; data curation, T.-N.T.; writing—original draft preparation, A.-C.P.; writing—review and editing, T.-N.T. and T.-C.P.; visualization, T.-N.T.; supervision, T.-C.P.; project administration, A.-C.P. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data are available on request by contacting the corresponding author.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Premier Healthcare Database being used by National Institutes of of Health to Evaluate Impact of COVID-19 on Patients Across the U.S. 2020. Available online: https://www.premierinc.com/newsroom/press-releases/premier-healthcare-database-being-used-by-national-institutes-of-health-to-evaluate-impact-of-covid-19-on-patients-across-the-u-s (accessed on 20 March 2022).
2. Chung, B.I.; Leow, J.J.; Gelpi-Hammerschmidt, F.; Wang, Y.; Del Giudice, F.; De, S.; Chou, E.P.; Song, K.H.; Almario, L.; Chang, S.L. Racial disparities in postoperative complications after radical nephrectomy: A population-based analysis. *Urology* **2015**, *85*, 1411–1416. [CrossRef] [PubMed]
3. Cheung, H.; Wang, Y.; Chang, S.L.; Khandwala, Y.; Del Giudice, F.; Chung, B.I. Adoption of robot-assisted partial nephrectomies: A population-based analysis of US surgeons from 2004 to 2013. *J. Endourol.* **2017**, *31*, 886–892. [CrossRef] [PubMed]
4. Chung, K.J.; Kim, J.H.; Min, G.E.; Park, H.K.; Li, S.; Del Giudice, F.; Han, D.H.; Chung, B.I. Changing trends in the treatment of nephrolithiasis in the real world. *J. Endourol.* **2019**, *33*, 248–253. [CrossRef] [PubMed]
5. Alavi, M.; Leidner, D.E. Knowledge management and knowledge management systems: Conceptual foundations and research issues. *MIS Q.* **2001**, *25*, 107–136. [CrossRef]
6. Alavi, M.; Leidner, D. Knowledge management systems: Issues, challenges, and benefits. *Commun. Assoc. Inf. Syst.* **1999**, *1*, 7. [CrossRef]
7. Gallupe, B. Knowledge management systems: Surveying the landscape. *Int. J. Manag. Rev.* **2001**, *3*, 61–77. [CrossRef]
8. Suchanek, F.M.; Weikum, G. Knowledge bases in the age of big data analytics. *PVLDB* **2014**, *7*, 1713–1714. [CrossRef]
9. Begoli, E.; Horey, J. Design principles for effective knowledge discovery from big data. In Proceedings of the 2012 Joint Working IEEE/IFIP Conference on Software Architecture and European Conference on Software Architecture, Helsinki, Finland, 20–24 August 2012 ; pp. 215–218.
10. Dong, X.; Gabrilovich, E.; Heitz, G.; Horn, W.; Lao, N.; Murphy, K.; Strohmann, T.; Sun, S.; Zhang, W. Knowledge vault: A web-scale approach to probabilistic knowledge fusion. In Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, New York, NY, USA, 24–27 August 2014; pp. 601–610.
11. Tretiakov, A.; Whiddett, D.; Hunter, I. Knowledge management systems success in healthcare: Leadership matters. *Int. J. Med. Inform.* **2017**, *97*, 331–340.
12. Maramba, G.; Coleman, A.; Ntawanga, F.F. Causes of Challenges in Implementing Computer-Based Knowledge Management Systems in Healthcare Institutions: A Case Study of Private Hospitals in Johannesburg, South Africa. *Afr. J. Inf. Syst.* **2020**, *12*, 4.
13. Manogaran, G.; Thota, C.; Lopez, D.; Vijayakumar, V.; Abbas, K.M.; Sundarsekar, R. Big data knowledge system in healthcare. In *Internet of Things and Big Data Technologies for Next Generation Healthcare*; Springer: Cham, Switzerland, 2017; pp. 133–157.
14. Le Dinh, T.; Phan, T.C.; Bui, T. Towards an architecture for big data-driven knowledge management systems. In Proceedings of the 22nd Americas Conference on Information Systems (AMCIS 2016), San Diego, CA, USA, 11–14 August 2016; Association for Information Systems: Atlanta, GA, USA, 2016.
15. Bierly, P.E.; Kessler, E.H.; Christensen, E.W. Organizational learning, knowledge and wisdom. *J. Organ. Chang. Manag.* **2000**, *13*, 595–618. [CrossRef]
16. Le Dinh, T.; Rickenberg, T.A.; Fill, H.G.; Breitner, M.H. Enterprise content management systems as a knowledge infrastructure: the knowledge-based content management framework. *Int. J. e-Collab. (IJeC)* **2015**, *11*, 49–70. [CrossRef]
17. Le Dinh, T.; Van, T.H.; Moreau, É. A Knowledge Management Framework for Knowledge-Intensive SMEs. In Proceedings of the 16th International Conference on Enterprise Information Systems ICEIS (3), Lisbon, Portugal, 27–30 April 2014; pp. 435–440.
18. Chambers, B.; Zaharia, M. *Spark: The Definitive Guide Big Data Processing Made Simple*, 1st ed.; O'Reilly Media, Inc.: Newton, MA, USA, 2018.
19. Singh, P. Natural Language Processing. In *Machine Learning with PySpark: With Natural Language Processing and Recommender Systems*; Apress: Berkeley, CA, USA, 2019; pp. 191–218.
20. Kreps, J.; Narkhede, N.; Rao, J. Kafka: A distributed messaging system for log processing. *Proc. NetDB* **2011**, *11*, 1–7.
21. Gates, M. *Machine Learning: For Beginners—Definitive Guide For Neural Networks, Algorithms, Random Forests and Decision Trees Made Simple*; CreateSpace Independent Publishing Platform: North Charleston, SC, USA, 2017.
22. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. In Proceedings of the 28th International Conference on Neural Information Processing Systems, Montreal, QC, Canada, 7–12 December 2015; pp. 91–99.
23. Girshick, R. Fast r-cnn. In Proceedings of the IEEE International Conference on Computer Vision, Washington, DC, USA, 7–13 December 2015; pp. 1440–1448.
24. Szegedy, C.; Ioffe, S.; Vanhoucke, V.; Alemi, A.A. Inception-v4, inception-resnet and the impact of residual connections on learning. In Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, San Francisco, CA, USA, 4–9 February 2017.
25. American Heart Association. *What Is High Blood Pressure?* South Carolina State Documents Depository: Washington, DC, USA, 2017; pp. 1–2.
26. Chobanian, A.V.; Bakris, G.L.; Black, H.R.; Cushman, W.C.; Green, L.A.; Izzo, J.L., Jr.; Jones, D.W.; Materson, B.J.; Oparil, S.; Wright, J.T., Jr.; et al. Seventh report of the joint national committee on prevention, detection, evaluation, and treatment of high blood pressure. *Hypertension* **2003**, *42*, 1206–1252. [CrossRef] [PubMed]
27. Ly, N.L.; Dong, V.H. *Brain Injury*; Medical Publishing House: HaNoi, Vietnam, 2013.