Building cloud-based healthcare data mining services

Citation:

This is the accepted manuscript.

©2016, IEEE

Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

The final definitive version has been published at: https://doi.org/10.1109/SCC.2016.66

Available from Deakin Research Online:
http://hdl.handle.net/10536/DRO/DU:30083661
Building cloud-based healthcare data mining services

Peng Zhang, Shang Hu, Jing He, Yanchun Zhang, Xinyu Cao, Guangyan Huang, Jiekui Zhang

Abstract—The linkage between healthcare service and cloud computing techniques has drawn much attention lately. Up to the present, most works focus on IT system migration and the management of distributed healthcare data rather than taking advantage of information hidden in the data. In this paper, we propose to explore healthcare data via cloud-based healthcare data mining services. Specifically, we propose a cloud-based healthcare data mining framework for healthcare data mining service development. Under such framework, we further develop a cloud-based healthcare data mining service to predict patients future length of stay in hospital.

Index Terms—cloud service, healthcare data, data mining, predictive model, FLOS

I. INTRODUCTION

Recent years have witnessed rapid growth of digital data in the modern healthcare industry. These data are mainly stored and isolated in disparate local systems, and are underutilized in terms of data analysis and knowledge discovery. Cloud computing techniques have made computational infrastructure capable of handling such enormous information burst in a cost-effective way. Up to the present, most works focus on migrating healthcare IT system and data storage to the cloud platform rather than taking advantage of information hidden in the data. As shown in Fig. 1, a typical healthcare cloud computing system has a hierarchical structure including system layer, control layer, and service layer. System layer constructs fundamental storage and computing environment using distributed computing resources, storage resources, and network resources. In control layer, system administrators control the load balancing, monitor system performance, and build programming environment. Finally service layer is responsible for providing large-scale healthcare services via real time management, privacy protection, and data analysis.

Extensive studies has been conducted in healthcare informatics, and data mining technologies have been exploited to solve healthcare related problems [1]. Although the prospect of healthcare data mining is intriguing, practical issues impede widespread adoption of data mining technologies in the healthcare industry:

- Healthcare data are heterogeneous in semantics and domain-specific meanings. Population-level data required by healthcare data mining are scattered across disparate and isolated local systems.
- The existing IT infrastructure and systems used by healthcare providers are mainly designed for administrative and billing purposes rather than intensive computing and service hosting. Developing large-scale healthcare data mining services requires huge investment in infrastructure.
- The development of workable healthcare data mining service requires expertise of various domains. Professionals from various domains should work in a well organized collaborative manner. No party can accomplish this task alone.
- Task-specific healthcare data mining solutions have been developed in the healthcare industry [1]. However, these solutions reside on isolated proprietary systems, and are developed for internal use only. The healthcare industry on the whole can benefit little from this endeavour. Moreover, these solutions lack in portability, scalability, and accessibility.

A. Related work

Electronic health records (EHRs) [2] are digitized health records collected from various health care settings. By means
of EHRs, medical information can be more conveniently stored and shared through clouds. Given the heterogeneity of distributed EHR system, it is important to develop a unique representation of medical entities and relationships between them. An ontology-based architecture [3] was proposed to integrate heterogeneous medical terms. Another ontology-based semantic model [4] was proposed to enable unified representation of multimodal health care data.

Instead of storing EHRs in isolated systems, health care cloud computing system stores EHRs in cloud storage [5]. Health care cloud computing system requires higher level of secure access control and performance than traditional cloud computing systems. A layered framework [6] was proposed to securely manage data. The authors also discussed how to build trusted applications from untrusted components. With the development of e-health, solutions for protecting privacy of patients and personal information are required urgently. Many methods achieve secure access control by exploiting encryption-based techniques [7] [8]. They generally encrypt confidential information or attributes and deliver the key to proper users. To fill the gap between client platform security and e-health system, a secure architecture for protecting confidential data of patients in client platform was proposed. To achieve secure access control of health care cloud computing, ESPAC (Efficient and Secure Patient-centric Access Control) [9] scheme was proposed; this scheme manages the access privileges of different users via a role-based access control model.

Data mining techniques have been applied to health care domain lately. An overview [10] of data mining applications within health care indicates major applied areas such as evaluating effectiveness of treatment, managing customer relationship, detecting fraud and abuse in health care economies, identifying chronic disease and high-risk patients, etc. A discussion [11] was given on current limitation and future research direction of EHRs mining. Diagnosing disease is one common application of EHRs mining; a recent work [12] uses data mining tools to recognize appropriate treatment for patients suffering from heart disease; another work [1] gives a survey on application of traditional data mining algorithms in health care fields, such as fraud prevention, detecting abnormal patients, mining association rules between unhealthy behaviors and diseases. Undiscovered correlations between diseases can be revealed by mining EHRs. Moreover, if we integrate EHRs data with genetic data we can obtain better understanding of underlying risk factors of certain diseases.

As citizen and government pay more attention to health care, the cost of medical services has raised tremendously. A recent study conducted by PricewaterhouseCoopers found that while the U.S. spends 2.2 trillion dollars each year on healthcare, nearly half of this amount is wasted [13]. One major part of such waste is attributed to potentially avoidable hospitalizations (PAHs) [14]. The Australia Institute of Health and Welfare (AIHW) define PAHs as “admissions to hospital that could have potentially been prevented through the provision of appropriate non-hospital health services” [15]. Most commonly, PAHs are related to chronic conditions such as diabetes complications, chronic obstructive pulmonary disease, gastroenteritis, and dental conditions etc. If patients who are at high risk of future PAHs can be early identified, the healthcare providers can develop individually tailored disease management plans for them before emergencies actually occur [16]. By receiving preventive treatments, healthcare interventions, and regularly scheduled follow-ups, certain disease development can be prevented [15]. In this way, unnecessary hospital admission can be avoided, and improved healthcare outcomes can be achieved with less healthcare expenditure. Intuitively, patients who have been admitted for hospital inpatient care are more likely to be re-admitted in the following years, especially when these patients have chronic diseases. More often than not, these patients have more comprehensive healthcare data which can reflect their current health conditions. If we can accurately predict these patients’ future length of stay (FLOS) in hospital based on their existing healthcare data, the predicted FLOS can be used directly as a quantitative indicator of their risk of future PAHs.

In this paper, we propose to explore healthcare data via cloud-based healthcare data mining services, specifically:

- we propose a cloud-based healthcare data mining framework under which various kind of cloud-based healthcare data mining services can be developed;
- we analyze the future LoS prediction problem from both data mining and cloud computing perspectives. Based on such analysis, we select most suitable data mining techniques and develop a cloud-based healthcare data mining service to predict patients future LoS.

B. Paper organization

The rest of this paper is organized as follows. In Section II, we introduce the proposed cloud-based healthcare data mining framework. In Section III, we analyze the FLOS prediction problem and develop solutions using data mining techniques. We conduct experiments on real world dataset and analyze the results in IV. Finally, Section V concludes our work.

II. CLOUD-BASED HEALTHCARE DATA MINING FRAMEWORK

Briefly speaking, building a healthcare data mining service requires investment in three aspects as illustrated in Fig.2:

- **Data**: healthcare data need to be collected, stored, integrated, and prepared for data mining process.
- **Infrastructure**: storage, computation power, operating system, runtime environment, and networks are required to support the development, deployment, and operation of data mining services.
- **Expertise**: medical expertise, data mining expertise, software engineering expertise, system administration expertise, and other IT-related expertise are all demanded to ensure practical workability of healthcare data mining services.

In this paper, we propose a cloud-based healthcare data mining framework to meet the requirements mentioned above, and facilitate the development, deployment, and the adoption
of healthcare data mining services. The structure of our proposed healthcare data mining framework is illustrated in Fig.3.

Under our proposed cloud-based healthcare data mining framework, heterogeneous patients’ health-related data previously distributed across disparate local data sources are integrated by the data integration module and stored in a central cloud-based repository. The database management system might be distributed, and the data might be physically stored in distributed geographical locations. But from the perspective of data access and data processing, this cloud-based data repository is centralized. The technical details of the data management and storage are transparent to the data scientists and software engineers. After the healthcare data integration process, invalid data are removed, redundant data are merged, and these original raw data are transformed into well structured standardized EHRs.

The EHRs obtained from data integration module cannot be directly used. The reasons lie in two folds. Firstly, the size of the data is too big. The EHRs contain exhaustive information regarding various aspects of people’s health condition. And the data mining tasks are very specialized focusing on specific problems. Secondly, the format of the EHRs might not suit the data mining techniques. These EHRs are derived from data generated from healthcare facilities, while the format was not designed for data mining tools. Therefore, data preparation process need to be conducted to extract a task-specific dataset for the data mining task. This process is conducted by the data preparation module. During this process, depending on the task and the data, various data processing techniques and data mining techniques are employed to generate a meaningful and concise dataset. Healthcare experts’ domain knowledge plays a critical role in this process. The medical knowledge can greatly facilitate understanding of the data, and make the data preparation process more effective and efficient. The domain knowledge and the power of data mining techniques are complementary. Utilizing both of them, task-related features are extracted from the data and the EHRs are projected onto the new feature space to generate a task-specific dataset.

Given task-specific dataset, data mining techniques are employed to analyze the data and construct data mining models in the model construction module. Here a data mining model is more than an algorithm implementation or a metadata description, but rather a set of data structures, statistics, and patterns that can be applied to any unobserved data to generate inferences and predictions. It represents the knowledge and insight learned from the data, and can be used to solve specific tasks, such as diagnostic pattern recognition, anomaly detection, classification, and numerical prediction, etc. There is no panacea algorithm that can solve all data mining problem. Building a data mining model is a data-driven, computation-intensive, and trial-and-error process. Data mining algorithm need to be selected based on both theoretical and experimental analysis. Model parameters need to be fine-tuned to achieve optimum performance. And the performance of the model need to be carefully assessed both analytically and empirically. We will further elaborate this part in Section III.

Data mining models provide core data mining functions, but they can not be conveniently used by end users. Therefore, the constructed models will be integrated in software solutions. These software solutions will be deployed as cloud applications or web-based application by the application deployment module. The resulting applications can be accessed by clinicians or patients via user-friendly interfaces.

Our proposed cloud-based healthcare data mining framework is generic, and various healthcare data mining services can be developed on the platform layer. These services will be deployed on the software layer and be accessible on a pay-per-use basis or via a subscription mechanism. The advantages of our proposed framework can be described from the following aspects:

- **Data:** Heterogeneous healthcare data collected and stored in disparate sources are integrated to provide a unified view to the data mining process. Individual healthcare providers act as both data contributors and data con-
sumers. They contribute their own share of data and consume the whole integrated data via data mining services.

- **Infrastructure & platform**: Healthcare data mining services are developed and deployed in a cloud computing platform. The cloud computing platform, consisting of operating system, program runtime environment, database management system, and web server, are delivered and managed by cloud provider. The underlying cloud infrastructure including hardware and software layers are transparent to data mining service development, and are provisioned on an on-demand basis. The cost and complexity of purchasing and managing the infrastructure can be avoided.

- **Service development**: In our proposed framework, the service development process is modularized. In this way, the development process can be better coordinated. Professionals specialized in various domains can focus on their own work while collaborating in a well organized manner. The modularization also makes update and maintenance easier.

- **Service accessibility**: Healthcare data mining services are deployed as cloud application softwares. Users can access the services via either dedicated cloud client software or simply web-based user interface. In this sense, the service can be accessed anywhere using various Internet enabled devices such as desktop computers, laptops, tablets, and smartphones, etc.

### III. Predictive Model of FLOS

Although our proposed healthcare data mining framework is generic, healthcare data mining services are very task-specific and hence need to be highly customized. Characteristics of the data structure, objectives of the tasks, and business issues all need to be taken into account so that suitable data mining techniques can be employed to achieve high performance and low cost. In this section, we propose a theoretical framework to analyze the FLOS prediction problem. Based on the analysis, we derive suitable data mining techniques to build predictive models as the core of a service.

Using the data mining language, predicting the patients FLOS is a regression problem, and hence can be described as follows:

Let the training dataset be denoted as $D = \{ (x_i, y_i) | i = 1, 2, \ldots, n \}$, where any $x_i = (x_{i1}, x_{i2}, \ldots, x_{id})$ is a data sample of the multivariate random variable $x = (x_1, x_2, \ldots, x_d)$ in the $d$-dimensional feature space $X$, and the corresponding $y_i$ is a sample of the scalar random variable $y$ in the target space $Y$. Given $D$, the goal is to learn a function $f^*$ from the space of function $\mathcal{H}$ such that the expected error $\mathbb{E}[E(f^*)] = \mathbb{E}[E(f^*(x), y_i)] = \mathbb{E}[E(y_i)]$ over $X \oplus Y$ is minimized.

In the above description, $\mathcal{H}$ is the function space containing all functions, $E(\cdot)$ is an error function, each component in the multivariate random variable $x = (x_1, x_2, \ldots, x_d)$ represents a feature describing one aspect of the patients health condition in the previous years, and the scalar random variable $y$ represents the length of stay in the following year. Using the training dataset, data mining algorithms learn a model which essentially is a function $f$. It takes any sample of the multivariate random variable $x$ as input, and outputs a scalar as the predicted FLOS.

Now the model learning problem can be described as an optimization problem: to find $f^*$ such that $f^* = \operatorname{argmin}_{f} \mathbb{E}[E(f)]$. Three problems immediately arise:

- the representational problem: the space of function $\mathcal{H}$ is infinitely large for searching, and how to represent a function in such space;
- the statistical problem: the feature space $X$ potentially has infinite number of data points, and the training dataset only contains finite number of instances;
- the computational problem: how to numerically solve the optimization problem in a systematic way.

Firstly, to solve the representational problem, instead of searching for $f^*$ in the space of all functions $\mathcal{H}$, we search for $f^*_X$ in a subspace of $\mathcal{H}$: $\mathcal{F}$. Here, $\mathcal{F}$ consists of a whole family of parameterized functions $f_X$. Secondly, to solve the statistical problem, instead of evaluating the expected error $\mathbb{E}[E(f)]$, we evaluate the empirical error $\mathbb{E}_{D}[E(f)]$ over the training dataset $D$. Thirdly, the computational problem is solved by using parameterized family of functions, various optimization techniques, tweaking model parameters, and applying domain-specific heuristics.

From the above discussion, $f^*$ is the best possible function among all functions. And $f^*_X$ is the best function in a smaller space of function $\mathcal{F}$, which indicates that $f^*_X = \operatorname{argmin}_{f} \mathbb{E}_{f}[E(f_X)]$. We further define $f^*_{X,D}$ as the best possible function we can achieve in $\mathcal{F}$ by evaluating the empirical error over the training dataset $D$, which means $f^*_{X,D} = \operatorname{argmin}_{f} \mathbb{E}_{f,D}[E(f_X,D)]$. Here we use $f_{X,D}$ to denote the functions which are in $\mathcal{F}$ and are candidates for $f^*_X$ under the empirical error evaluation scheme. These functions form a new space of function $\mathcal{F}'$ which is a subspace of $\mathcal{F}$ since using empirical error essentially shrinks the searching scope within $\mathcal{F}$. Fig. 4 depicts the three spaces of functions and the three optimum functions.

As depicted in Fig.4, the outer closed solid curve denotes the space $\mathcal{H}$ which contains all functions. The optimum function in $\mathcal{H}$ is denoted as $f^*$ which can be considered as fixed for a given data mining task. The inner solid curve denotes
the space $\mathcal{F}$, which is a subspace of $\mathcal{H}$. $\mathcal{F}$ is determined by data mining model. Once a data mining model is selected, $\mathcal{F}$ is fixed. Given a particular data mining model and a fixed training dataset, the inner dashed curve denotes the effective searching space $\mathcal{F}^*$ within which the data mining algorithm searches for $f_{\mathcal{F}, D}$. Since only $\mathcal{F}^*$ is the effective searching space, the optimization objective of the data mining task can be replaced by minimizing $\mathcal{E} = \mathbb{E}_D[\mathbb{E}[f_{\mathcal{F}, D}]]$.

Taking into account the ideal minimum expected error $\mathbb{E}[\mathbb{E}[f^*]]$, the model-dependent minimum expected error $\mathbb{E}[\mathbb{E}[f]]$, and the minimum empirical error $\mathbb{E}_D[\mathbb{E}[f_{\mathcal{F}, D}]]$, $\mathcal{E}$ can be decomposed into four terms as in Equation (1):

$$
\begin{align*}
\mathcal{E} &= \mathcal{E}_{\text{app}} + \mathcal{E}_{\text{est}} + \mathcal{E}_{\text{opt}} + \mathbb{E}[\mathbb{E}[f^*]] \\
\mathcal{E}_{\text{app}} &= \mathbb{E}[\mathbb{E}[f^*]] - \mathbb{E}[\mathbb{E}[f]] \\
\mathcal{E}_{\text{est}} &= \mathbb{E}_D[\mathbb{E}[f_{\mathcal{F}, D}]] - \mathbb{E}[\mathbb{E}[f]] \\
\mathcal{E}_{\text{opt}} &= \mathbb{E}_D[\mathbb{E}[f_{\mathcal{F}, D}]] - \mathbb{E}_D[\mathbb{E}[f_{\mathcal{F}, D}]]
\end{align*}
$$

Firstly, the approximation error $\mathcal{E}_{\text{app}}$ quantifies the representational problem. It measures how closely a model can represent or approximate any function. Conceptually, it reflects the “gap” between $\mathcal{H}$ and $\mathcal{F}$. Secondly, the estimation error $\mathcal{E}_{\text{est}}$ quantifies the statistical problem. It measures how closely we can use $f_{\mathcal{F}, D}$ in the data-constrained function space $\mathcal{F}^*$ to approach $f^*$ in the model-dependent function space $\mathcal{F}$. Conceptually, it reflects the “gap” between $\mathcal{F}$ and $\mathcal{F}^*$. Thirdly, the optimization error $\mathcal{E}_{\text{opt}}$ quantifies the computational problem. Once the data mining model is chosen and the training dataset is fixed, the effective searching space $\mathcal{F}^*$ is fixed. However, the optimization process might not be able to find $f_{\mathcal{F}, D}$ within $\mathcal{F}^*$, but rather end up at some local optima.

According to Equation (1), since $\mathbb{E}[\mathbb{E}[f^*]]$ can be viewed as fixed for any given data mining task, minimizing $\mathcal{E}$ equals minimizing the sum of the approximation error $\mathcal{E}_{\text{app}}$, the estimation error $\mathcal{E}_{\text{est}}$, and the optimization error $\mathcal{E}_{\text{opt}}$. Ideally, if we can simultaneously reduce $\mathcal{E}_{\text{app}}$, $\mathcal{E}_{\text{est}}$, and $\mathcal{E}_{\text{opt}}$, then we can reduce the overall error $\mathcal{E}$:

- To reduce the approximation error $\mathcal{E}_{\text{app}}$ we need to select a more flexible model which can represent a larger family of functions. Hence its corresponding function space $\mathcal{F}$ is larger, the gap between $\mathcal{H}$ and $\mathcal{F}$ is reduced, and more likely the approximation error $\mathcal{E}_{\text{app}}$ can be reduced.
- Given a flexible model, to reduce the estimation error $\mathcal{E}_{\text{est}}$ we need to employ a larger training dataset which can better represent the true data distribution in the feature space, and support the complex models. Hence the gap between $\mathcal{F}$ and $\mathcal{F}^*$ is reduced, and more likely $\mathcal{E}_{\text{est}}$ can be reduced.
- With flexible model and large dataset, $\mathcal{F}^*$ becomes larger and more complex. Therefore, we need better optimization mechanism and more computation power to perform optimization and reduce $\mathcal{E}_{\text{opt}}$.

In practice, it is hard to simultaneously reduce $\mathcal{E}_{\text{app}}$, $\mathcal{E}_{\text{est}}$, and $\mathcal{E}_{\text{opt}}$: 1) some models are very flexible such that they can asymptotically represent any function [17]. But the optimal training of these models has been proved NP-hard [18]; 2) we only have data of limited size, which cannot support infinitely complex model; 3) with very complex model and very large dataset, the optimization becomes expensive and even intractable using existing optimization methods and computational resource. Therefore, a trade-off need to be achieved among model complexity, available data, and optimization.

Based on the above theoretical analysis, in this paper, we choose to use decision-tree-based ensemble learning method to build predictive model for patients FLOS. Ensemble learning method constructs a set of base-models and combine them to build an ensemble of models to achieve synergy. More often than not, the ensemble of models can achieve higher performance than a single model [19]. Specifically, we choose to use decision-tree-based ensemble learning method. The advantages of using such method are as follows:

- Decision tree is a very flexible model. Theoretically, given sufficient training data, decision tree can asymptotically approximate any function [17].
- Typical healthcare data have various data type: categorical, ordinal, and numerical. Decision tree supports all these data types and is invariant to scaling of inputs.
- Decision tree is a flexible and unstable learning algorithm, which makes diverse and accurate base-models as required by ensemble learning method.
- Using decision tree as the base-model in ensemble learning makes the implementation easier, model tuning easier. And the final model and result are simple to understand and interpret.
- Ensemble learning method can reduce $\mathcal{E}_{\text{app}}$, $\mathcal{E}_{\text{est}}$, and $\mathcal{E}_{\text{opt}}$ in an elegant manner. The base-models are deliberately built simple, which makes the optimization easier.
- Ensemble learning method by nature is highly modularized. The model building process can be carried out in a parallel manner, which is a desirable feature in the cloud computing setting.
- The predictive model produced by tree-based ensemble method is separate from the training dataset, which means once the model is constructed there is no need to keep the training dataset in storage. The model only requires very small memory and can process prediction queries very fast in a cost-effective manner. These features make tree-based ensemble model perfect in the pay-per-use cloud computing setting.

A. Random Forest

Random Forest (RF) [20] is an ensemble algorithm which builds a set of decision trees using random sampling over the data records and features. To construct a decision tree for classification, each internal node splits the dataset by evaluating the information gain. To construct a regression tree, the concept of information gain can be replaced by variance reduction to accommodate the regression need.

Given a training set $D$ and feature set $\mathcal{F}$, at each internal node we want to learn a splitting rule $\theta_i = \{f_i, a_i\}$ which indicates that: partitioning the feature space into two parts by the hyperplane $f_i = a_i$. Here $f_i$ denotes a specific feature, and $a_i$ is a constant value $f_i$ takes. Among all possible splitting rules $\Theta = \{\theta_i | i = 1, 2, ..., \}$, we choose the one which maximize the information gain, and training data go
to different branches according to such splitting rule. The information gain \( g(D, \theta) \) is defined as in Equation (2).

\[
g(D, \theta) = H(D) - H(D|\theta)
\]

\[
H(D) = -\sum_{k=1}^{K} \frac{|C_k|}{|D|} \log_2 \frac{|C_k|}{|D|}
\]

\[
H(D|\theta) = \sum_{i=1}^{n} \frac{|D_i|}{|D|} H(D_i)
\]

In which \( K \) is the number of total classes and \( n \) is the number of total branches, \( |D| \) is the number of all training samples, \( |C_k| \) is the number of training samples with class \( k \) and \( |D_i| \) is the number of training sample in each branch. Obviously \( H(D) \) is the entropy of training set \( D \), and \( H(D|A) \) is the entropy of partitioned training set with splitting rule \( \theta \).

The information gain is the difference between \( H(D) \) and \( H(D|A) \). We can recursively build a decision tree at each internal node with splitting rule that has the largest information gain. Based on this principle, there are several variants of algorithms includes ID3, C4.5, CART (Classification And Regression Tree) and so on. The structure of RF is illustrated in Fig .5.

![Random Forest diagram](image)

**Fig. 5. Construction of Random Forest models.**

As shown in Fig. 5, the training process of RF includes the following steps:

1. Independently and randomly sample the original training dataset with replacement for \( k \) times to generate \( k \) smaller training dataset. Each smaller training dataset is used to independently construct a decision tree.
2. During the building of each individual decision tree, at each internal node, randomly select a smaller set of features \( F' \) from \( F \) and only consider features in \( F' \) rather than all features in \( F \) to construct splitting rules.
3. After all base-model decision trees are constructed, assemble their output using weighted vote.

The advantages of Random Forest can be described from three aspect: the representational aspect, the statistical aspect, and the computational aspect corresponding to the three problem we discussed in previous paragraphs.

- **Representational:** In RF, each tree is designed to be built simple, which avoid intractable complexity. While simple tree has less representational power, RF employs ensemble of multiple diverse trees to increase the overall model’s representational power.
- **Statistical:** Randomness is injected into RF. The injected randomness give rise to diversity among the base-models.
- **Computational:** Decision trees in RF are built simple deliberately, which makes the optimization easier. By averaging multiple decision trees, the ensemble might avoid local optima. Moreover, Each tree in RF is constructed independently, each node at the same depth of the same tree is splitted independently, and each feature is evaluated independently for splitting. Therefore, the whole model can be constructed in a highly parallel manner, which is a great advantage in the cloud computing setting.

B. **Gradient Boosting Machine**

Gradient Boosting Machine (GBM) [21] is another ensemble learning method. It sequentially constructs a series of simple trees rather than independently as in Random Forest. A new tree is constructed to reduce the error made by the existing trees. Each tree will be assigned a weight which is the optimum setting to reducing the overall error. In this way, the constituent trees forms an ensemble. The construction process of GBM is demonstrated in Fig .6.

![Gradient Boosting Machine diagram](image)

**Fig. 6. Construction of Gradient Boosting Machine models.**

The GBM algorithm can be described as follows:

1. Initializing the model learning as in Equation (3).

\[
f_0(x) = \arg \min_{c} \sum_{i=1}^{n} L(y_i, c)
\]

2. For \( m = 1, 2, ..., M \):
   2.1 For \( i = 1, 2, ..., n \), computing the pseudo-residuals \( r_{m,i} \):

\[
r_{m,i} = \frac{\partial L(y_i, f(x_i))}{\partial f(x_i)}
\]

2.2 Choosing a suitable learner \( p_m(x) \) for residual \( r_{m,i} \).
2.3 Computing multiplier \( c_m \):

\[
c_m = \arg \min_{c} \sum_{i=1}^{n} L(y_i, f_{m-1}(x_i) + c \cdot p_m(x_i))
\]

2.4 Updating

\[
f_m(x) = f_{m-1}(x) + c_m \cdot p_m(x)
\]

3. Return final model \( f_M(x) \).

In the above pseudo code, \( M \) is the number of iterations, \( n \) is the number of training samples, \( r \) is the empirically
estimated residual which is decreasing fastest at the gradient direction of loss function and $c$ is the multiplier which represents the importance of the current learner.

In GBM algorithm, each tree is constrained to be simple, which helps control the complexity of the ensemble and facilitate optimization. The gradient of the error function is employed to continue the optimization process along the whole series of trees, which can achieve better optimization performance and reduce the optimization error $E_{opt}$. The trees are constructed sequentially rather than independently as in RF, and each succeeding tree is correcting the predecessors and greedily adjust the overall model. In this sense, the GBM model has more representational power, which helps reduce the approximation error $E_{app}$. Furthermore, as a tree-based model, a GBM model can also be constructed in a parallel manner which fits the cloud computing setting very well.

IV. EXPERIMENTAL RESULTS AND ANALYSIS

In this section we test the Random Forest and Gradient Boosting Machine algorithm on a real world dataset\(^1\). This dataset contains patients’ historical claims data. And the goal is to predict the patients’ FLOS in the following year. The raw attributes of these EHRs are briefly depicted in Table I.

<table>
<thead>
<tr>
<th>MemberID</th>
<th>Sex</th>
<th>AgeAtFirstClaim</th>
<th>Provider</th>
<th>Vendor</th>
<th>PrimaryCare</th>
<th>Specialty</th>
<th>PlaceOfService</th>
<th>PayDelay</th>
<th>LengthOfStay</th>
<th>DaysSinceClaim</th>
<th>PrimaryCondition</th>
<th>CharlsonIndex</th>
<th>ProcedureGroup</th>
<th>Year</th>
<th>DaysSinceService</th>
<th>Drug</th>
<th>DrugCount</th>
<th>Lab</th>
<th>DaysSinceService</th>
<th>LabCount</th>
</tr>
</thead>
</table>

After data preparation process as proposed in [22], the final dataset consists of 139 features and around 147,000 records. We split this dataset into 117,000 records in training dataset $D$ and around 30,000 records in testing dataset $T$. To evaluate the effect of training dataset size on the performance of data mining algorithm, we conduct 20 rounds of experiments for both RF and GBM. In each round, we independently sample 5% more data from the full size training dataset $D$ to form a new training dataset $D_i$, $i = 1, 2, ..., 20$. We establish a benchmark by using existing cloud-based machine learning service\(^2\). The error function used to evaluate the models’ performance is defined in Equation (6).

$$R = \sqrt{\frac{1}{n} \sum_{i}^{n} \left[ \log(p_i + 1) - \log(v_i + 1) \right]^2}$$

(6)

In the above error function, $p_i$ is the predicted FLOS and $v_i$ is the real length of stay, $n$ is the total number of records. The experimental result is illustrated in Fig. 7.

\(^1\)www.heritagehealthprize.com
\(^2\)aws.amazon.com/machine-learning/

In Fig.7, the result of Linear Regression is obtained by conducting the experiment with full training dataset using Amazon Machine Learning service. This test result is used as a benchmark. As Fig.7 illustrates, RF and GBM can achieve better performance using much smaller training dataset.

By increasing the size of training dataset, both RF and GBM can achieve reduced error. When the training dataset is small, RF can obtain better performance than GBM. However, as the available training data increase, GBM outperforms RF. According to the error decomposition in Equation (1) and our analysis in previous section, RF is more effective in reducing the estimation error $E_{est}$, but less effective in reducing $E_{app}$ and $E_{opt}$. In contrast, GBM is more effective in reducing $E_{app}$ and $E_{opt}$, but less effective in reducing $E_{est}$. When the training dataset is relatively small, the estimation error is dominant. Hence RF can obtain better performance. As the size of available training dataset grows, the approximation error and optimization error become more critical. GBM has more representational power and better optimization mechanism. Hence, with larger training dataset, GBM can achieve better performance than RF. The distribution of the real hospital length of stay is demonstrated in Fig. 8. And the distribution of the GBM predicted FLOS is illustrated in Fig. 9. As shown in Fig. 9, most of predicted FLOS are between 0 and 1, which is very close to the real data distribution.

Given the theoretical analysis is section III, and the experimental analysis in this section, is it possible that we can combine the advantages of RF and GBM to construct a better model? Furthermore, can we utilize other data mining methods other than decision tree as the base-model and combine the base-models in other way to construct an ensemble? The answer is yes and the general method is called stacking [23]. Instead of using only one model, i.e. the decision tree, as RF and GBM do, stacking method adopts multiple models to construct multiple base learners and construct a multi-level ensemble over the base learners. Theoretically, the stacking method is difficult to analyze and there is no general accepted way of constructing a stacking-based ensemble. In practice, a stacking-based solution is often constructed in a highly customized manner and requires more task-specific engineering.
effort. Nevertheless, RF and GBM method can be integrated in stacking settings to achieve better overall performance [24].

Fig. 8. Distribution of real hospital length of stay.

Fig. 9. Distribution of GBM predicted FLOS.

V. CONCLUSION

In this paper, we proposed a cloud-based healthcare data mining framework. Under such framework, various cloud-based healthcare data mining services can be developed, deployed, and provisioned to the general healthcare industry for knowledge discovery and decision-making support. In our proposed framework, 1) population-level healthcare data scattered across disparate local data sources are integrated, which provides abundant data for the data mining process; 2) computational infrastructure and resources can be delivered by cloud computing platforms in a reliable, scalable, and cost-effective manner, which satisfies the computational and financial requirement for building healthcare data mining services; 3) the service development process is modularized, which makes the service development, update, and maintenance easier and faster; 4) the healthcare data mining services are deployed and provisioned to the healthcare practitioners as either cloud applications or web services, which ensures high service accessibility. Under the proposed framework, we analyzed the patients FLOS prediction problem in the context of data mining, and built data mining service to solve this problem. This data mining service can help the healthcare practitioners to better understand the data, make optimum clinical and administrative decision, and develop data-driven patient-centered healthcare services.

REFERENCES