IoT-Based Big Data Storage Systems in Cloud Computing: Perspectives and Challenges

Hongming Cai, *Senior Member, IEEE*, Boyi Xu, *Member, IEEE*, Lihong Jiang, *Member, IEEE*, and Athanasios V. Vasilakos, *Senior Member, IEEE*

***Abstract*—Internet of Things (IoT) related applications have emerged as an important field for both engineers and researchers, reflecting the magnitude and impact of data-related problems to be solved in contemporary business organizations especially in cloud computing. This paper first provides a functional framework that identifies the acquisition, management, processing and mining areas of IoT big data, and several associated technical modules are defined and described in terms of their key characteristics and capabilities. Then current research in IoT application is analyzed, moreover, the challenges and opportunities associated with IoT big data research are identified. We also report a study of critical IoT application publications and research topics based on related academic and industry publications. Finally, some open issues and some typical examples are given under the proposed IoT-related research framework.**

***Index Terms*—Big data, business intelligence, cloud computing, data management, distributed processing, Internet of Things (IoT) applications, performance isolation.**

I. Introduction

INTERNET of Things (IoT) technology have been an popular approach to implement and run business applications in the past years. Since massive data have been generated by huge amounts of distributed sensors, how to acquire, integrate, store, process and use these data has become an urgent and important problem for enterprises to achieve their business goals. As a consequence, both of researchers and engineers are faced with the challenge of handling these massive heterogeneous data in highly distributed environments, especially in cloud platforms. Referring to some related research [1], characteristics of IoT data in cloud platforms can be summarized as follows.

1) *Multisource High Heterogeneity Data:* IoT applications acquire data from different distributed sensors. These data types vary from integer to character, including semi-structured and unstructured data such as images, audio, and video streams. How to integrate these distributed data from multisource is fundamental for application development.
2) *Huge Scale Dynamic Data:* IoT applications always connect a huge quantity of sensors. Communications between different objects in a large-scale dynamic environment generate a large volume of real-time, high-speed, and uninterrupted data streams. Thus, scalable storage, filtering and compression schemes are essential for efficient data processing in cloud platform.
3) *Low-Level With Weak Semantics Data:* IoT data from sensors are of low-level with weak semantics before they are processed. Relations of these data are temporal-spatial correlation. For the purpose of execution of these intelligent systems, complex semantics need to be abstracted in event-driven perspective from the mass of low-level data.
4) *Inaccuracy Data:* Some experiments show that most sensing systems can only capture 1/3 correct data caused by unreliable reading, which brings difficulties into direct usage. Thus, multidimension data analysis and processing are important for wide adoption of IoT applications.

IoT-based data storage systems in cloud computing face three pairs of conflict requirements, which are distributed execution with united management of infrastructure resources, multitenant storage with isolated performance, and scalability with flexible. In addition, by the use of cloud platform for IoT data exchanging, processing and integration, different requirements are given for mass, real-time and unstructured data processing covering different levels, such as data representation, data storage, and data analysis.

Based on data processing function, this paper first provides a functional framework that identifies the acquisition, management, disposing and mining areas of IoT data. Several associated functional modules are defined and described in terms of their key characteristics and capabilities. Then, current research in IoT applications is analyzed to identify the challenges associated with related functional areas. Based on research analysis, some future technical tendencies are also proposed.

This paper is organized as follows. Section II gives a framework of IoT in which data-processing process is given to show the overview of related studies on the view of application. Next, we introduce studies that discuss related new technological developments and related

Manuscript received May 6, 2015; accepted October 6, 2016. Date of publication October 19, 2016; date of current version February 8, 2017. This work was supported by the National Natural Science Foundation of China under Grant 61373030 and Grant 71171132.

H. Cai and L. Jiang are with the School of Software, Shanghai Jiao Tong University, Shanghai 200240, China (e-mail: hmcai@sjtu.edu.cn; jiang-lh@cs.sjtu.edu.cn).

B. Xu is with the College of Economics and Management, Shanghai Jiao Tong University, Shanghai 200052, China (e-mail: byxu@sjtu.edu.cn).

A. V. Vasilakos is with the Department of Computer Science, Electrical and Space Engineering, Luleå University of Technology, SE-931 87 Skellefteå, Sweden (e-mail: th.vasilakos@gmail.com).

Digital Object Identifier 10.1109/JIOT.2016.2619369

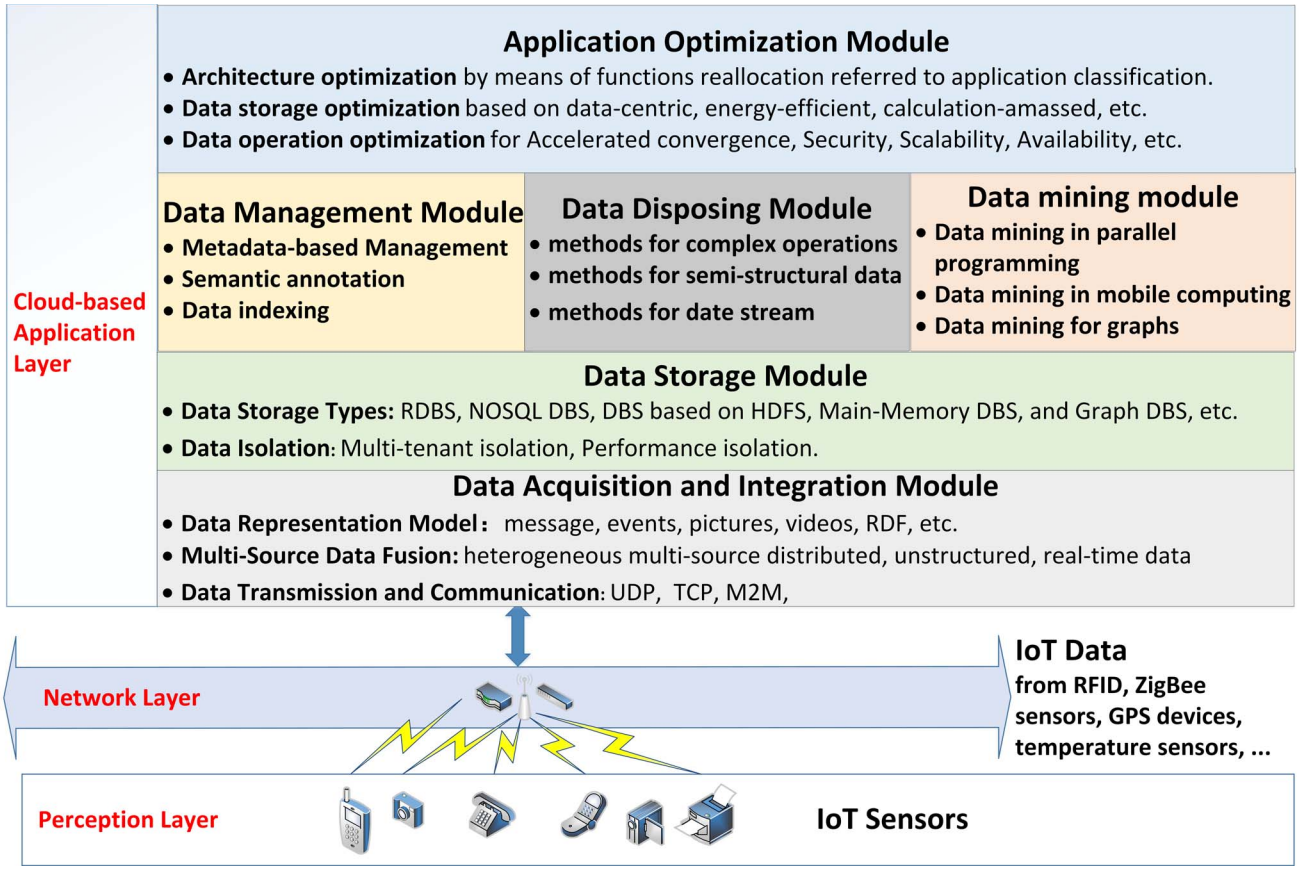

Fig. 1. Framework of IoT-based data storage systems in cloud computing.

performance consideration covering six areas in Section III. Then, Section IV proposes several open issues related to recent IoT developments and applications. Finally, Section IV concludes our survey, highlights challenges, and provides an outlook.

II. Functional Framework of Cloud-Based IoT Applications

A common IoT application framework [2] consists of perception layer, network layer, and application layer. Application layer is critical for IoT-based storage systems in cloud computing because it is composed of middlewares and business models. Much work has been done to enable effective and intelligent data processing and analysis in application layer based on cloud computing. Front-end layer involves radio frequency identification (RFID), wireless sensor network (WSN), and other smart things. Based on the processing process of IoT application, a framework of IoT-based data storage systems in cloud computing is given, as Fig. 1 shows. The framework consists of several modules, which are data acquisition and integration, data storage, data management, data processing, data mining, and application optimization module.

Referring to Fig. 1, related technologies can be divided into several functional modules as follows.

1) *Data Acquisition and Integration Module:* As an input module, how to acquire and integrate heterogeneous data from distributed and mobile devices is a fundamental problem for the whole system construction.
2) *Data Storage Module:* Considering different types of IoT data, including structured, semi-structured, and unstructured data of huge quantity, different kinds of database or file system, such as XML files in Hadoop distributed file system (HDFS), relational database management system (RDBMS), and Not only SQL (NoSQL). An NoSQL database provides a mechanism for storage and retrieval of data that is modeled in means other than the tabular relations used in relational databases. Graph DBMS should be combined to achieve a high efficiency for data storage in cloud platforms.
3) *Data Management Module:* For the purpose of searching and retrieving data from huge volume of data sources with high efficiency, different approaches, such as data index, metadata, semantic relations, and linked data are realized for data management in different platforms.
4) *Data Processing Module:* In cloud platform, mass data processing mechanisms, such as MapReduce are constructed for parallel and distributed data processing. Data querying and reasoning can be carried out in a more flexible way to adapt to large volumes of data.
5) *Data Mining Module:* Considering that data from sensors are always raw and low-level data, high-level information needs to be extracted, classified, abstracted and analyzed for application purpose. Thus, data mining

on IoT data mainly aims to achieve comprehensive views or data analysis results for end-users.

6) *Application Optimization Module:* Based on application analysis, related algorithms or approaches are required for processing IoT data in cloud platform providing different performance requirements, such as decreased I/O, accelerated convergence, security, scalability, availability, management, decreased cost and price, etc.

III. Methods and Challenges

Based on the above framework of IoT data processing process, related research are divided in the following sections to show current methods and challenges.

A. Data Acquisition and Integration Module

Data acquisition and integration module is designed to get data from different types of sensor devices, such as RFID, ZigBee sensors, GPS devices, temperature sensors, etc. Heterogeneous data brings a big challenge to IoT applications when the developers need to integrate massive structured, semi-structured, and unstructured data. From the perspective of data processing in computing, we can classify the three main methods in the process of data acquisition and integration: data representation models, multisource data fusion, and data transmission and communication.

1) Data Representation Models: Data Representation models are used for IoT data acquisition and integration fundamentally. There exist several different types of sensor devices, such as messages, events, pictures, videos, status data, etc. Data representation models should be designed based on different application purposes with a flexible and common format.

Traditional transmitting method for the sensor data in proprietary formats is not enough for the IoT applications because different objects of embedded systems are different either in the grammar or context level. Therefore, event data are required to manage in a unified, integrated and correlated way because the event data rely on massive, diverse, and interrelated data sources. Therefore, sensor data must be enriched and transformed into resource description framework (RDF) format for further processing. In [3], a framework called HEP which integrates the representation of relational and XML event streams is proposed to support a unified event fusing and processing with a general specification. The framework could be used to express almost all types of windows so as to understand easily and keep a good scalability. In [4], a corresponding virtual object model is proposed to enrich those sensors with context information so as to support smart-city application based on cognitive IoT objects. The model represents real objects (sensors) for mass data remotely accessing, and produces a stream of raw sensor measurements.

User interface is becoming increasingly important with the development of the IoT applications. For the purpose of abstract sensing and actuation primitives of smart devices, a model-based interface description scheme [5] is proposed to generate user interface. The user interface description languages can carry out intuitive interfaces for developers to propose taxonomy of device controller. Hasan and Curry [6] proposed an approach to derive semantic similarity and relatedness on a distributional statistical model of semantics. A rule language and an event matcher is used to present an approximate model so as to divide the semantic coupling dimension for further processing. This model has been validated correctly by large sets of events from real-world smart city. On consideration of amassed heterogeneous data streams, an event information management platform [7] is proposed to collect and analyze data streams coming from heterogeneous sensors. The platform makes applications run as cyber-physical-social systems.

In the field of semantic level interaction with context, there are some existing research and standards. SensorML, which is an approved Open Geospatial Consortium standard, provides standard models and an XML encoding for describing sensors and measurement processes. And the World Wide Web Consortium (W3C) has initiated the Semantic Sensor Networks Community Group to develop the semantic sensor network (SSN) ontology. The SSN ontology can model sensor devices, systems, processes, and observations objects so as to enable expressive representation of sensors, sensor observations, and knowledge of the environment. The SSN ontology is encoded in the Web Ontology Language and has begun to achieve broad adoption and application within the sensors community. It is currently being used by various organizations, from academia, government, and industry, for improved management of sensor data on the Web, involving annotation, integration, publishing, and search.

2) Multisource Data Fusion: Heterogeneous data from multiple sources mean the data from different sensors have different structures. When heterogeneous and various sensor data are acquired, multisource data should be merged to create a comprehensive and meaningful view for further utility. This data integration process is also called multisource data fusion to unify data from different data sources.

Considering heterogeneous data from different sensor devices makes the aggregation, integration and collaboration of the data harder, a heterogeneous data integration model [8] is proposed for data integration and analysis. The model provide aim to eliminate distributed data heterogeneity as well as build a customized application view for the upper applications. And also a way to maintain the integrity and consistency of the data. In the level of protocol, a conceptual solution for heterogeneous sensor data integration [9] in crowd sensing applications is presented to combine different kinds of protocols for different types of data. Three kinds of protocols, such as HL7 for medical data, BACnet for building monitoring, and observation and measurement model for environmental data, are integrated into one data model to manage sensors and actions. Aim to work in a heterogeneous network, a novel approach [10] is proposed to monitor a typical plastic industry environment with WSN, services and Google Gadgets. It uses microinjection to make use of a heterogeneous network of wireless sensor nodes and these sensor nodes transmit environment data, such as material storage conditions, ambient lab temperature, and humidity. In [11], IoT resources are integrated as a novel automatic resource type on the business process layer so as to accommodate more changes in future enterprise

environments. The research solves the problem of the integration of the IoT paradigm and its devices coming with native software components as resources. Thing Broker [12] aims to integrate IoT objects with different characteristics, protocols, interfaces, and constraints while maintaining the simplicity and flexibility required for a variety of applications. Thing Broker provides a uniform access interface to different IoT objects. Using a single abstraction to represent IoT objects with their own configurable attributes, Thing Broker involves all sorts of objects, from physical sensors to high-level services.

3) Data Transmission and Communication: Once data acquisition from different sensors is finished, data transmission and communication should be carried out so as to transmit data to the back-end or to communicate with other sensors for business purpose.

There are many protocols for data transmission and communication, such as user data protocol (UDP) and TCP. Developers often use UDP to transmit multimedia data due to its real-time characteristic. But when network congestion and channel noise occur, packet loss happen easily by UDP protocol. A new real-time multimedia transmission protocol over UDP called control over UDP (CoUDP) [13] is designed to solve this problem. The performance of CoUDP protocol is better than UDP and TCP because it adds rate control and fast retransmission mechanism over UDP application and gives up redundant feedback like TCP. In [14], a layered fault management scheme with fault managing program control and separate layer functions is designed to ensure the reliability of end-to-end transmission for IoT applications. This proposed scheme suits well to the IoT requirements. They use fuzzy cognitive maps to realize integrated evaluation and prediction of the possible fault, which can solve the problem that current relative algorithms are not suitable for the complex conditions. In [15], a proximity-based authentication approach is proposed to utilize the wireless communication interface. And the advantage of the approach is that inferring proximity within about 1 s relies on ambient radio signals.

In short, current methods related to data acquisition and integration could be included as follows: for the reason of the great diversity and heterogeneity of data, data presentation models in cloud environment do not have a unified form in the aspect of sensor data acquisition and integration. Multisource data fusion is still a main problem in applications. Although data transfer protocols are mature, there are still some problems in the interoperability between sensor data.

B. Data Storage Module

In IoT applications, the massive data from sensors consume large storage space. Meanwhile, because that different roles and tenants require different service and security levels, data should be isolated for various requirements. Therefore, how to share and isolate these data in cloud platform are the main challenges in IoT data storage.

1) Data Storage Types: In the aspect of data storage in cloud platform, existing works can be classified into several types as: RDBMS, NOSQL DBMS, DBMS based on HDFS, main-memory DBMS, and graph DBMS.

a) RDBMS: Many structured data storage platforms are based on RDBMS. Although massive data is generated rapidly and variously, the relations between these data are always essential for a multitenant data storage system [16]. Then different traditional relational data with virtual relational data is combined in a single schema, but exports a unified data access view to act as a multitenant database for different tenants. There is an approach of RDBMS called Ultrawrap [17], which encodes a logical representation of each RDBMS as an RDF graph, and uses SPARQL queries to get the data on the existing relational stored views.

b) NoSQL DBMS: Unlike RDBMS, NoSQL DBMS stores and manages unstructured data in a key-value model. The NoSQL DB is free in schema structure. It can provide some properties, such as horizontal scalability, distributed storage, dynamically schema, etc. On the other hand, NoSQL DB is not good at keeping atomicity, consistency, isolation, and durability of data. Besides, it cannot support well for some distributed queries. Besides, on the aspect of database access, some work has been done on the potential integration of an ontology-based date access approach in NoSQL stores [18]. It is a new data management style that exploits the semantic information represented in ontologies when searching the IoT data stores. Therefore, some related work has tried to integrate the feature of distributed file system to the IoT data stores.

c) DBMS integrated with HDFS: HDFS can also be extended to a special distributed file repository, which processes massive unstructured files efficiently. In the area of IoT data stream, many data are generated in the XML format, and how to deal with these small-sized, huge-volume XML files becomes an important challenge. One approach [19] is to optimize storing and accessing massive small XML files in HDFS. Small XML files are merged into a larger file to reduce the metadata at name node, thus related mechanism could be used to improve the data store performance. With the help of a new central-indexing service discovering system [20], based on Hadoop HBase data store, the performance of service discovering is increased.

d) Main-memory DBMS: High performance of IO stream processing is important for large-scale IoT application. Lu and Ye [21] implemented a large-scale RFID application in the main-memory database system H2. Besides, it also provides a multidimensional hash-based index design framework and achieves an outperformed performance evaluation. Hara *et al.* [22] analyzed the physical database structure that realizes high-speed database migration, which is an important part of the cloud data storage, and propose several recovery approaches to the migratory main-memory databases.

e) Graph DBMS: Graph DBMS is a database that uses graph structures with nodes, edges, and properties to represent and store data. A database that uses graph structures for semantic queries with nodes, edges and properties to represent and store data. With graph DBMS, the relationship among sensor data can be managed efficiently. Reference [23] provides a high performance Graph DBMS management system, supporting efficient manipulation of large graphs that consists of large-scale nodes and edges.

TABLE I
COMPARISON BETWEEN DATA STORAGE TYPES

Features	RDBMS	NOSQL DBMS	DBMS integrated with HDFS	Main-Memory DBMS	Graph DBMS
Support for ACID	Not well	Yes	Yes	Common	Common
Support for semi-structured data and unstructured data	Not well	Yes	Yes	Yes	Use graph structures with nodes, edges
Support for structured data	Yes	Not well	Yes	Yes	Use graph structures with nodes, edges
Support for scalability	Not well	Yes	Yes	Not well	Yes
Support for massive and distributed processing	Not well	Yes, but not flexible	Yes	Yes, but not flexible	Yes

f) RDF-based data storage: RDF is a semi-structured data model for Web information resources management. RDF schema (RDFS) [24] provides an ontology denoting language for grouping the resources into concepts and identifying the relationship among these concepts. Cloud-based RDF data management [25] provides a principled categorization of existing work on RDF data management.

Aiming to support different data type of IoT sensors, different data type should be combined so as to realize effective data storage. A comparison of different data storage types is given in Table I.

2) Data Isolation in Cloud Computing: Another challenge in IoT data storage comes from resource elasticity in the IoT cloud computing framework. The difference of data authority and data provision performance arise the requirement of data isolation in cloud platform. For the purpose of improving the utilization of resource, most cloud platform allows the tenants to share the same computing resources. This approach may cause the problem of inconsistency and latency in data content, and low efficiency in data performance.

a) Multitenant isolation: In the view of storage realization, the common multitenant data isolation methods can be classified into four types as shared table pattern, dependent table pattern, dependent database pattern, and dependent virtual machine pattern. The shared table pattern in [26] reduces the cost on switching between different hosted owners. The dependent table pattern, dependent database pattern, and dependent virtual machine pattern mean that different tenants have their own target table, database or virtual machine.

Cai *et al.* [27] proposed a database management model that supports multiple database integrating and unified data accessing. Besides, the model also integrates a multitenant data isolation mechanism. And in [27], a layered reference model is proposed for IoT data management. The model consists of data cleaning layer, event processing layer, data storage, and analysis layer. It provides a layered view of the data management in IoT area.

b) Performance isolation: For resource provision, a tenant-based resource allocation model for resource management in cloud computing environment is desperately needed. Some related works have been done on this area. It provides formal measurements for provisioning of virtualized resources in cloud environment [28], and provides a resource allocation model and a multitenant configuration environment to the application on the cloud platform. In order to maintain data consistency, [29] designed a distributed caching and scheduling middleware-MicroFuge which can provide performance isolation for storage system. Based on an empirical-driven performance model, MicroFuge uses an adaptive deadline-aware cache eviction module to reduce the deadline misses. Similar idea is also used in noninvasive and energy efficient performance isolation in virtualized servers. The framework-NINEPIN combines self-adaptive machine learning and a robust target tracking predictive model [30]. It outperforms a good performance isolation approach on a virtualized server cluster.

In general, aim to adapt to high heterogeneity of IoT data from distributed data sources, it is a popular way to combine different data types, such as RDBMS integrated with HDFS, so as to construct a scalable data storage in cloud environment. However, there is still great contradiction between user authority and performance flexibility. Performance isolation has to be implemented in different levels with a consideration of different data types. The problem of sharing and isolating these data in cloud platform is still a main challenge in IoT data storage on considering characteristics of different applications.

C. Data Management Module

Data management module forms an intelligent and effective database for further distributed or parallel IoT application. In the aspect of data management, related works can be classified into three aspects: 1) metadata management; 2) semantic annotation; and 3) data indexing.

1) Data Management Based on Metadata: Metadata management is generally defined as the end-to-end process and the governance framework for creating, controlling, enhancing, attributing, defining, and managing a metadata schema, model or other structured aggregation system, either independently or within a repository and the associated supporting processes. It can make data easily organized and understood by users without being involved with everything concerning the accessing solution.

Many storage systems, such as HDFS, decouple metadata management from file data access. Based on HDFS in cloud computing, [31] proposes a metadata management scheme. This scheme employs multiple name nodes and divides metadata into "buckets," which can be migrated among name nodes dynamically according to the workloads of system. Also, in order to maintain reliability, metadata is replicated in different

name nodes with log replication technology. Paxos algorithm is used to keep replication consistency.

In [32], an efficient distributed metadata management scheme is proposed for cloud computing. It can deliver high performance and scalable metadata services by using techniques of metadata distribution method based on parent directory path ID, mimic hierarchical directory structure, improved Chord, cooperative double layer cache mechanism algorithm, and the application of database to metadata. In order to access the distributed data with reduced latency, the dynamic metadata model in database for cloud computing [33] is proposed to reduce the overhead problem occurred in the time that the data from the data server is retrieving.

Automatic metadata generation using associative networks [34] provides an automatic metadata generation system that leveraged resource relationship generated from existing metadata as a medium for propagation from metadata-rich to metadata-poor resources. By means of a discrete-form spreading activation algorithm, metadata associated with metadata-rich resources is propagated to metadata-poor resources for different substrate of associative network.

2) Semantic Annotation: With the rapid development of the cloud computing, users expect a better experience through cloud computing, such as service computing and multimedia computing. Lots of IoT applications heavily depend on the understanding of the data, so that large-scale data annotation has received intensive attention in recent years. As an important part of retrieval technology, the accuracy of semantic annotation determines the retrieval results.

A new service model is proposed in [35], which is closely related to previous distributed computing methods, such as Web services and grid computing. By mean of two function encapsulation covered the services registering with the semantic description and the services searching with accomplish the required expectations, the researchers implement a semantically enhanced platform to assist the process of cloud service discovery.

In [36], a new approach is presented to semantic annotation with linked data for document enrichment in the domain of education. Differed from traditional semantic annotation which connect relevant term of the document with an instance of the ontology, the approach connects relevant terms to graphs of the ontology. And after its expansion process, only relevant and contextualized information is included. Since the document is annotated with a set of interconnected graphs, students can access and navigate through these contents in the document so as to deepen the topics. This approach provides a better description, moreover considering the semantic nature of linked data and is more suitable in the domain of education.

Tao *et al.* [37] presented a scheme for image annotation on the cloud, which transmits mobile images compressed by Hamming compressed sensing to the cloud and conducts semantic annotation through a special support vector machine on the cloud.

3) Data Indexing Strategy: Data indexing can make data retrieval operations more effective at the cost of additional writing operations and extra data storage spaces for data indexes. With indexes, DBMS can quickly locate data without searching every row in the database tables every time when database tables are accessed. In the area of data indexing, existing related work can be classified into three types that are bitmap index, complex data structure index, and inverted index.

a) Bitmap index: Bitmap index is a kind of database index that uses the bitmap data structure. Bitmap index is considered to work well for low-cardinality columns and is suitable for analytical processing with less data storage space. To reduce the load of retrieving process in cloud storage, a method of bucketization [38] is proposed based on the traditional bitmap index. The tuples with attributes of interest are divided into buckets, depending on the attribute values that they choose, and then the original attribute values will be hidden according to the relevant bucket indexes. And in [20], in order to help to trace objects along supply chains in the area of IoT, Li *et al.* [20] proposed storage schema which uses event time-stamp to identify column and event index content to serve as cell value to build central-indexing device service system.

b) Complex data structure index: Due to the weakness of the index of bitmap data structure in transaction processing, index types of other data structure are used to improve the performance. Ma *et al.* [39] used B+ Tree and R-Tree to make efficient indexes for the massive data of IoT in cloud computing environment. Also, in [40], an object-store called Walnut is proposed, which is developed at Yahoo! and uses the bLSM index based on the special data structure and LSM-Tree for various cloud data management systems.

c) Inverted index: An inverted index is a kind of index data structure that stores a mapping from content to its locations in order to improve full text searching. It is the most popular data structure used in document retrieval systems for large-scale search engines. Indexing word sequences for ranked retrieval [41] aims to present and analyze a new index structure designed to improve query efficiency in dependency retrieval models. They have presented a novel approach to estimating n-gram statistics for information retrieval tasks. The index structure is scalable in both query processing time and space requirements. The index structures describe the exploration of n-grams as query features. In [42], a data scheme that extends the inverted index approach is proposed to combines with techniques for the design of SSE. To reduce the size of the inverted indexes, Vishwakarma *et al.* [43] proposed an approach to prune the whole document from the index based on its importance and relevance of top-k results. The elimination is taken based on the scores of individual documents.

A comparison of different data index methods is given in Table II.

With the rapid increase in the amount of data and their correlation, automatic metadata generation, ontology generation and evolution, and efficient, low cost and dynamic-updating data indexing have attracted great attention. How to implement a unified pervasive data management model that solves the contradiction between secured sharing and performance isolation is the emphasis and difficulty in current study of data management.

TABLE II
COMPARISON BETWEEN DATA INDEXING METHODS

Features	Bitmap Index	Complex Data Structure Index	Inverted Index
Data structure	Bitmap	Tree, graph or others	Mapping from content to location
Suitable data characteristics	Values of a variable repeat frequently	Values of a variable repeat frequently	New key values monotonically increase, such as sequence numbers
Suitable scene	Analytical process, such as OLAP	Transaction process, such as OLTP	Large scale process, such as search engines
Performance	Less efficient	Efficient	More efficient
Cost	Less space	More space	Most space and increased processing
Current usage status	Common	Common	Rising trend

D. Data Processing Module

In cloud-based parallel or distributing data processing, MapReduce [44] and its open source implementation Hadoop is one of the most popular parallel processing methods in cloud platform. For the purpose of parallelization, scalability, load balancing, and fault-tolerance, MapReduce is widely used in query processing for data analysis tasks in cloud platforms. However, MapReduce does not directly support more complex operations such as joins. More research on high-level, declarative management of complex data such as RDF is required for massively parallel processing of IoT data in the cloud.

1) Parallel Processing Methods for Complex Operations: In [45], a processing framework called wave is designed for bulk data processing, incremental computing, and iterative processing. Framework wave uses an implicit mechanism to synchronize the parallel programs execution without any user specification on which programmers use events and trigger reactions to process the data. The selective embedded just-in-time specialization (SEJITS) [46] executes complex analytic queries on massive semantic graphs in big-data analytics. A domain-specific language is implemented to enable flexible filtering and customization of graph algorithms without sacrificing performance, using SEJITS selective compilation techniques.

2) Parallel Processing Methods for Semi-Structural Data: For the purpose of RDF data processing, an efficient and customizable data partitioning framework SPA [47], which targets at distributing processing of big RDF data, is presented to support fast processing of different size as well as complexity. A MapReduce framework [48] is designed to carry out SPARQL query processing. Thus, RDFS reasoning can be involved in deductive databases and therefore recursive query processing techniques are implemented.

3) Parallel Processing Methods for Data Stream: In cloud platform, sometimes data arrives in stream and the processing algorithm is tasked with data without explicitly storing it. Existing parallel frameworks in cloud, such as MapReduce and its variations are unable to support complex parallel processing effectively. Traditional algorithms of sequential pattern may raise the scalability challenge when dealing with big data. To address the problems of optimizing parallel data mining, a heuristic cloud bursting algorithm, maximally overlapped binpacking driven bursting, is developed which considers the time overlap to improve data mining parallelization [49]. Spreitzer *et al.* [50] presented Ripple, a middleware that is built on iterated MapReduce for distributed data analytics with the support of various styles of analytics in the same platform and on the same data.

In general, distributed processing in cloud environment is still mainly based on MapReduce, which can be conducted after the expansion of different type (structured, semi-structured, and unstructured) data. However, on consideration of some demerits of MapReduce, such as high communication cost, redundant processing, and lack of interaction ability in real-time processing, the methods on high-performance distributed data processing without MapReduce are required in some application related to complex processing like approximate reasoning or high interactive processing.

E. Data Mining Module

Moreover, although there are many data mining methods employed in cloud computing, we cannot just transplant the algorithms directly into IoT application. Due to the high dynamic and wide distributed features of IoT data, it is needed to find an effective and efficient way to process huge amount of data. There are mainly three kinds of data mining for IoT applications.

1) Data Mining in Parallel Programming: Data mining algorithms based on cloud platforms are mainly in parallel pattern in contrast to other platform. The research [51] focuses on classification problem in data mining area in cloud platform, which is motivated from the identification of birds, and it propose its own modern cyclic approach to solve the classification problem which is verified to be quite efficient. And [52] takes more interest in clustering problem, and proposes a parallel K-means algorithm based on Hadoop platforms. Others may find it more important to mine the association rules of big data or do some predictions based on the mining results. Yu and Zhu [53] proposed a dynamic resource-provisioning algorithm to predict resource utilization.

2) Data Mining in Mobile Computing: Mobile computing is another hot topic in recent years, and may be another direction of development for IoT data analysis since IoT also involves plenty of smart devices. The research on data mining in mobile environment can be considered as a good reference for IoT data mining. The research [54] proposes a novel algorithm named wireless heterogeneous data mining (WHDM) to find the frequent patterns or knowledge from big data and WHDM, which is proved efficient. In [55], distributed Hoeffding trees is used to classify streaming data in mobile computing environment. Personal mobile commerce pattern mining and prediction [56] develops pattern mining

TABLE III
COMPARISON BETWEEN DATA MINING METHODS

Data Mining Dimensions	Goals	Shortcomings or Future Work
parallel programming	Classification	Need more experiments in different environments
	Clustering	Need to verify the results about the efficiency and convergence
	Association Rules	Need to optimize the algorithm for smaller data sets or nodes
	Prediction	Only focus on one specific area
Mobile Computing	Frequent patterns, belongs to association rules	Lack of comparison experiments
	Data stream classification	Have some outliers to fix
	Survey on clustering problem	Focus on one specific area
Graph Mining	BSP-based Parallel Graph Mining (BPGM) based on BSP computing model	Limits on the scale of graph data
	cloud-based SpiderMine (c-SpiderMine) based on cloud computing	Examine more real big data sets and introduce more data mining algorithms
	Graph mining on MapReduce	Improvements on efficiency

and prediction techniques that explore the correlation between the moving behavior and purchasing transactions of mobile users.

3) Data Mining for Graphs: Among different events, some exhibit strong correlations with the network structure, while others do not. Such structural correlations will shed light on viral influence existing in the corresponding network. Unfortunately, the traditional association mining concept is not applicable in graphs because it only works on homogeneous data sets like transactions and baskets. A special kind of data mining, graph mining is usually used to explore the frequent patterns from networks or databases. The special structure of graph makes it distinctive in data mining. In addition, it is usually more efficient in some specific areas compared to common cloud data mining method. Chen *et al.* [57] proposed a new approach called the cloud-based SpiderMine that uses cloud computing to do graph mining. And similarly, Lai *et al.* [58] also make use of cloud computing and propose a robust and efficient MapReduce-based graph mining tool. The research [59] proposes a novel measure for assessing structural correlations in heterogeneous graph data sets with events.

Considering the different methods of IoT data mining in cloud platform, a comparison is given in Table III.

In general, the mass IoT data on the cloud platform are highly dynamic and time-related. Most of them are formed as dataflow. Traditional data mining methods are unable to intelligently process the data. This is the weakest as well as the most difficult link in current applications. Dataflow oriented processing technology for real-time intelligent application are the main challenges of IoT data mining in cloud platform.

F. Comprehensive Application Optimization Module

The efficiency optimization of application layer can be viewed from three aspects as follows.

1) Architecture Optimization: Based on application analysis, architecture optimization is an effective way to achieve good performance. In [60], an efficient scheduling model with caching mechanism is proposed for the gateway of distributed sensors in smart-living. Peng *et al.* [61] analyzed the characteristics of data transmission, and proposed a message oriented middleware data processing model in IoT so as to make data transmission more efficient and more convenient. By using machine learning-based analysis, a middleware approach is implemented [62] to link scenario information to network and derive meaningful communication predictions for efficient wireless communication.

2) Data Storage Optimization: Wang *et al.* [63] proposed a storage approach that uses hierarchical extended storage mechanism to handle massive dynamic data. It can store data separately according to the data type and add storage nodes dynamically. Data duplication is a process that breaks data streams into some smaller data chunks and removes duplicate chunks. Tan *et al.* [64] pointed out that the removal of duplicate data chunks leads to a de-linearization of data placement, which sometimes affects the read performance, throughput and efficiency. An effective way is proposed to cut down the de-linearization of data placement, which requires little reduction on compression ratios. IoT data is often stored in the shape of small files. Dong *et al.* [65] proposed different approaches to improve the efficiency of storing and accessing small files on cloud storage, which include file grouping or file merging and prefetching schemes. Ning *et al.* [66] designed a new virtualized socket library with the technology of shared memory in data transmission, which uses a buffer to store I/O requests.

3) Data Operation Optimization: Because the effectiveness and scalability of MapReduce-based implementations of complex data-intensive tasks depend on an even redistribution of data between map and reduce tasks, complex redistribution approaches are necessary to achieve load balancing among all reduce tasks to be executed in parallel for skewed data. Kolb *et al.* [67] proposed two approaches for skew handling and load balancing to reduce the search space of entity resolution, utilize a preprocessing MapReduce job to analyze the data distribution, and distribute the entities of large blocks among multiple reduce tasks. The current practices of static slot configuration and fair resource sharing may not efficiently utilize resources. When high priority jobs are sharing resource with lower priority jobs, fair sharing is against priority-based scheduling. P2P-MapReduce [68] provides a more reliable MapReduce middleware that can be effectively applied in dynamic cloud infrastructures. To provide a more reliable MapReduce middleware, an adaptive MapReduce framework named P2P-MapReduce, is designed

which exploits a peer-to-peer model to manage node churn, and provides a good fault tolerance level. Lu *et al.* [69] studied the optimization of resource utilization in Hadoop and present a nonintrusive slot layering solution that uses two tiers of slot (active and passive) to increase the degree of concurrency with minimal performance interference.

The optimization of IoT applications can be conducted from many aspects. It is currently a feasible and important method to divide the applications into compute-intensive and storage-intensive types and optimize-related algorithms, respectively, on the basis of architecture optimization, which attracts much attention. However, the architectural features of different kinds of applications are varied from different levels of data contents or storage types.

IV. New Technology in Future Prospects

From the past ten years, we are stepping from Web1.0 which focus on single direction information creation and passive customer to Web2.0 which attracts information co-creation and active customer stage. Now, we are moving from Web2.0 toward Web3.0, the stage of the ubiquitous computing Web.

Based on characteristics analysis of IoT data in Section I, we provide some open issues referred to different technical areas on handling these challenges, which covered new architecture for the integration cloud computing with IoT objects seamlessly, smart things for contextual and real-time event processing mechanism, big linked data for massive semantic storage and management, new parallel and dynamic processing pattern for high performance, data stream mining for dynamic and uncertainly decision making and so as to describe future prospects of IoT storage systems in cloud platform.

A. Cloud of Things

With the rapid development of IoT application in cloud platform, number of connected devices has increased in a very high speed. It has been said that the devices are more than the people on the Earth in 2011. And the connected devices are expected to reach 24 billion by 2020. All this devices will connected via cloud platforms for different applications. IoT and cloud computing working in integration makes a new paradigm, which have been termed as cloud of things (CoT) [70].

In CoT, IoT objects are extended from sensors to every front-end things in the Internet. And distributed sites are connected as a whole body, such as smart house, smart factory, smart city, and smart planet. Based on CoT, a logical architecture of smart city [71] is given. With the merge of cloud platform and IoT, CoT is required to enrich the ability for massive devices interactive and interoperability so as to support smart and intelligent applications. The CoT will take more and more important role in different industries and research areas. Some issues such as resource allocation balanced energy and efficiency, IPv6-based identity management, quality of service provisioning, architecture for data storage, security and privacy and unnecessary communication of data will be involved in CoT [2]. And based on ontology-based multitenant data management in cloud environment, a typical CoT platform is also given for IoT applications [72], as shows in Fig. 2.

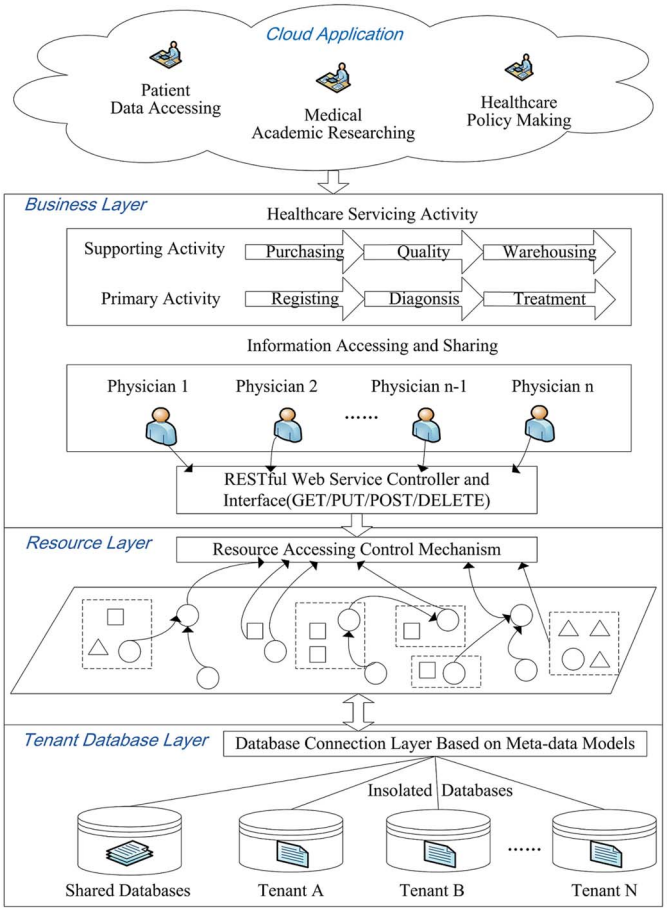

Fig. 2. Ontology-based CoT platform [72].

When the numbers of devices are increasing, heterogeneous data and services will be involved in CoT. Other than data and resources in single view of cloud or current IoT application, CoT will pay more attention in a business insight. The issues related to integration of IoT with cloud computing, require smart gateway to perform the rich tasks and preprocessing, which traditional sensors are not capable of fulfilling the task of data storage and processing.

B. Smart Things With Contextual Data Representation

Sensor data as a service faces the issues of interoperability and reusability for massive heterogeneous sensor data and data services. Therefore, how to integrate these data acquirement modules with flexible performance, especially, how to develop a smart device as an intelligent and self-organizational device in the cloud platform are important open issues.

Since connected devices are rapidly increasing, there will be lot of data as well. Massive unstructured and semi-structured data, such as video, image, and XML files are generated from the devices in IoT applications. So how to dispose these big data is the main challenge.

Considering that online object association could act as a user interface of smart things and provide a semantic

relationship for intelligent application, intelligence and contextualization model is important and fundamental on the purpose of construction a unified semantic view for further processing purpose. However, it is difficult to maintain consistence and archive intelligence in a massive heterogeneous data environment. From the view of data processing, a typical semantic case is Lilliput [73], which is an ontology-based platform for the IoT. By providing semantic information, including online social networks and contextual information, such as location of smart things as a social graph, Lilliput enables unified access control with less effort to support intelligent IoT application without having to understand device details with a good consistence.

An architecture supporting semantic data processing is given for knowledge acquisition and management [74]. The architecture aims to find high-level information from heterogeneous Web resources, such as RDF files. In it, the unstructured content analysis capabilities, such as UIMA are integrated in a coordinated environment supporting the processing, transformation and projection of produced metadata into RDF semantic repositories. Although current IoT semantic association is able to reflect up-to-date physical/online context, it has a limitation of representing temporal social relationship among people, objects, and places, thus yielding additional cost on utilizing various machine learning methods. With the development of intelligent IoT applications, enhanced intelligence and contextualization models would enrich IoT more expressivity semantic association and support social interaction reasoning between smart things. It will facilitate smart things to construct a convenient and powerful devices or environment for intelligent IoT applications.

C. Big Linked Data for Semantic Data Management

Considering complex data associations are generated from different sources or complex data structure, extracting relevant information in multilingual context from massive amounts of unstructured, structured and semi-structured data is a challenging task. Various theories have been developed and applied to ease the access to multicultural and multilingual resources.

Linked data [75] is defined as some typed links between data from different data sources, such as different databases or data nodes by means of the Web. Big linked data named Blinked data [76] is an instance of big data which is the union of big and linked data. For the purpose of effective data management, semantic annotation based on linked data provides a new issue in a massive, complex associated and contextual application scene. These associated and contextual data play a critical role for intelligent application.

Driven-by semantic technology such as linked data and ontology, we could predict semantic data processing approaches will get a great improvement in the near future. And a more natural and meaningful way with high-level information will be common in different IoT area. Combined with natural language processing, the semantic technology will be used to create more intelligent application. A referred linked data processing platform is also given for social networks (see Fig. 3).

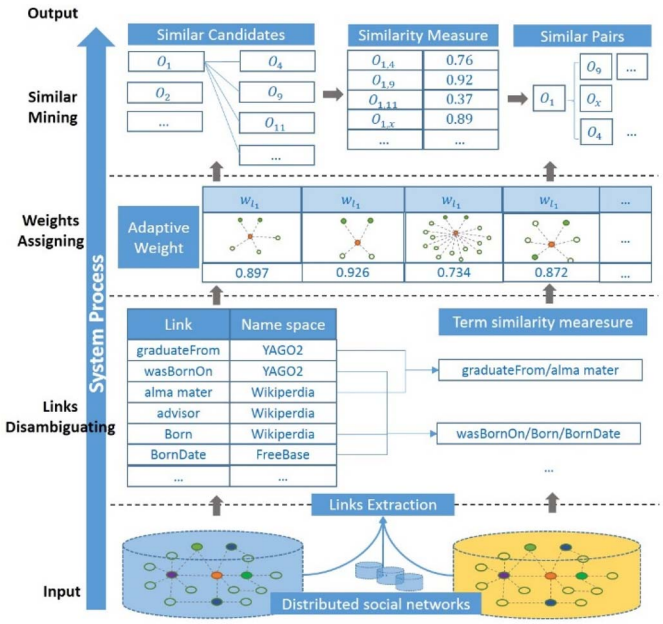

Fig. 3. Referred architecture of social networks application based on linked data [77].

D. Data Stream Mining for Intelligent Application

Unstructured data such as video data could not be stored into a structured database system for analysis purpose. And data mining on dataflow form different data sources with nonpersisted association is a new but important issue. There are several different directions to process dataflow with some dynamic methods, for example, to retracted features from continuous dataflow so as to build data association, or to process the whole body of a fragment of dataflow by function transformation. Data stream processing in dynamic and decentralized peer-to-peer network [78] process data streams of different active data sources. The approach involved three areas of data source management, continuous query distribution and distributed query management.

Data stream mining involved uncertain reasoning based on partition data and utility of intermediate result for high efficiency. When unstructured and semi-structured data are also involved in the processing process, there are lots of research and technical problem left to do.

E. New Pattern for Parallel and Dynamic Data Processing

In big IoT data environment, data are changing on types, state and analysis purpose. Other than centralized master-server implementations, parallel and particle data processing framework is need to enable the execution MapReduce pattern in dynamic cloud infrastructures. An approach which using the MapReduce framework for large-scale graph data processing is given [79]. The approach relies on a density-based partitioning to build balanced partitions of a graph database over a set of machines. The experiments show that the performance and scalability are satisfying for large scale of data processing.

Scale features of the big data on various parts in many rapidly changed sources produce obstacles to find useful information from these data. Therefore, these rebuild and

TABLE IV
WEAKNESSES IN MAPREDUCE AND SOLVING TECHNOLOGY FOR IOT BIG DATA DISPOSING

IoT Data features	Disposing Requirement	Weakness in MapReduce	Improved Techniques
Distributed multi-source high heterogeneity data	Data access	Selective access to data	Indexing, Data Layout, International Data Placement
	Data communication	High communication cost	Partitioning, Colocation
	Process n-way operation from multiple sources	Lack of support for n-way operations	Additional MR phase, Redistribution of keys, Record duplication
Huge scale dynamic data	Work allocation	Load balancing	Pre-processing sampling, repartitioning, batching
	Real-time data processing	Lack of interactive or real-time processing	Streaming pipelining, in-memory processing, pre-computation
	Data stream disposing	Redundant and wasteful processing	Result sharing, Incremental Processing, Batch Processing of Queries, Result Materialization
Low-level with weak semantics data	Interactive query in data analysis tasks	Sequence execution lack of interaction	Looping, Caching, pipelining, Recursion, incremental processing
	Query Optimization	lacks of management and future reuse of results	Processing optimizations, parameter tuning, plan refinement, operator reordering, code analysis, dataflow optimization
Inaccuracy data	Exploratory queries	Lack of quick retrieval of approximate result	Reasoning based on formal expression such as ontology
	rank-aware processing such as top-k queries	Lack of early termination	Sorting, Sampling

re-execution data mining algorithm are not applicable for big data analysis system. We need new dynamic data mining algorithms on the dataflow other than competed structural data.

However, despite its evident merits such as scalability, fault-tolerance, ease of programming, and flexibility, MapReduce has limitation in interactive or real-time processing on handling IoT data processing. MapReduce is not perfect for every large-scale analytical task, and the high communication cost and redundant processing make a big challenge for IoT application. In [80], a technical framework for improvement MapReduce is given. Based on weakness and current solved methods, we given an optimization requirement on IoT data for a large-scale processing purpose (see Table IV).

There exists an urgent need for design some new data processing patterns other than MapReduce pattern, we could predict that the next generation of parallel data processing systems for massive data sets should combine the merits of existing approaches to support complex operation and unstructured data in a parallel and dynamic way.

V. CONCLUSION

As the IoT technologies are evolving, a substantial amount of their applications have been founded in many industries. This paper is a timely research which overviews the current and potential IoT big data storage systems in cloud computing and at the same time surveys the state-of-art in literature from the view of data processing process.

The IoT storage system enables tracking of essential information about items as they move through cloud platforms. It shows significant value for IoT applications by providing an accurate knowledge of the current IoT data processing, which results in higher availability and flexible resource provision.

Data storage system supporting IoT devices can be utilized to improve the entire data processing efficiency and offer huge competitive advantage to the IoT applications. It has been shown that semantic relationships among IoT data will lead to greater global intelligent and interoperational capabilities (contextual business scene, semantic annotation, multidevices cooperation, etc.). IoT data storage systems will enable enterprise to acquire such capability.

REFERENCES

[1] M. Ma, P. Wang, and C.-H. Chu, "Data management for Internet of Things: Challenges, approaches and opportunities," in *Proc. IEEE Int. Conf. IEEE Cyber Phys. Soc. Comput. Green Comput. Commun. (GreenCom) IEEE Internet Things (iThings/CPSCom)*, Beijing, China, 2013, pp. 1144–1151.

[2] M. Aazam, I. Khan, A. A. Alsaffar, and E.-N. Huh, "Cloud of things: Integrating Internet of Things and cloud computing and the issues involved," in *Proc. 11th Int. Bhurban Conf. Appl. Sci. Technol. (IBCAST)*, Islamabad, Pakistan, Jan. 2014, pp. 414–419.

[3] W. Wang and D. Guo, "Towards unified heterogeneous event processing for the Internet of Things," in *Proc. 3rd Int. Conf. Internet Things (IOT)*, Wuxi, China, 2012, pp. 84–91.

[4] A. Somov, C. Dupont, and R. Giaffreda, "Supporting smart-city mobility with cognitive Internet of Things," in *Proc. Future Netw. Mobile Summit (FutureNetworkSummit)*, Lisbon, Portugal, 2013, pp. 1–10.

[5] S. Mayer, A. Tschofen, A. K. Dey, and F. Mattern, "User interfaces for smart things—A generative approach with semantic interaction descriptions," *ACM Trans. Comput. Human Interact.*, vol. 21, no. 2, 2014, Art. no. 12.

[6] S. Hasan and E. Curry, "Approximate semantic matching of events for the Internet of Things," *ACM Trans. Internet Technol.*, vol. 14, no. 1, 2014, Art. no. 2.

[7] M.-S. Dao *et al.*, "A real-time complex event discovery platform for cyber-physical-social systems," in *Proc. Int. Conf. Multimedia Retrieval*, Glasgow, U.K., 2014, p. 201.

[8] H. Liu, Y. Liu, Q. Wu, and S. Ma, *Geo-Informatics in Resource Management and Sustainable Ecosystem* (Communications in Computer and Information Science). Heidelberg, Germany: Springer, 2013, pp. 298–312.

[9] S. Villarroya *et al.*, "Heterogeneous sensor data integration for crowd-sensing applications," in *Proc. 18th Int. Database Eng. Appl. Symp.*, Porto, Portugal, 2014, pp. 270–273.

[10] U. Raza, B. Whiteside, and F. Hu, "An enterprise service bus (ESB) and Google gadgets based micro-injection moulding process monitoring system," in *Proc. IET Conf. Wireless Sensor Syst. (WSS)*, 2012, pp. 1–6.

[11] S. Meyer, A. Ruppen, and C. Magerkurth, "Internet of Things-aware process modeling: Integrating IoT devices as business process resources," in *Proc. Int. Conf. Adv. Inf. Syst. Eng.*, Valencia, Spain, 2013, pp. 84–98.

[12] R. A. P. de Almeida *et al.*, "Thing broker: A Twitter for things," in *Proc. ACM Conf. Pervasive Ubiquitous Comput.*, Zürich, Switzerland, 2013, pp. 1545–1554.

[13] W. Jiang and L. Meng, "Design of real time multimedia platform and protocol to the Internet of Things," in *Proc. IEEE 11th Int. Conf. Trust Security Privacy Comput. Commun.*, Liverpool, U.K., 2012, pp. 1805–1810.

[14] X. Li, H. Ji, and Y. Li, "Layered fault management scheme for end-to-end transmission in Internet of Things," *Mobile Netw. Appl.*, vol. 18, no. 2, pp. 195–205, 2013.

[15] H. Shafagh and A. Hithnawi, "Poster: Come closer: Proximity-based authentication for the Internet of Things," in *Proc. 20th Annu. Int. Conf. Mobile Comput. Netw.*, Maui, HI, USA, 2014, pp. 421–424.

[16] H. M. Yaish, M. L. Goyal, and J. G. Feuerlicht, "Multi-tenant elastic extension tables data management," *Proc. Comput. Sci.*, vol. 29, pp. 2168–2181, Jun. 2014.

[17] J. F. Sequeda and D. P. Miranker, "Ultrawrap: SPARQL execution on relational data," *Web Semant. Sci. Services Agents World Wide Web*, vol. 22, pp. 19–39, Oct. 2013.

[18] O. Curé, F. Kerdjoudj, D. Faye, C. L. Duc, and M. Lamolle, "On the potential integration of an ontology-based data access approach in NoSQL stores," *Int. J. Distrib. Syst. Technol.*, vol. 4, no. 3, pp. 17–30, 2013.

[19] Y. Zhang, W. Han, W. Wang, and C. Lei, "Optimizing the storage of massive electronic pedigrees in HDFS," in *Proc. 3rd Int. Conf. Internet Things (IOT)*, Wuxi, China, 2012, pp. 68–75.

[20] M. Li, Z. Zhu, and G. Chen, "A scalable and high-efficiency discovery service using a new storage," in *Proc. IEEE 37th Annu. Comput. Softw. Appl. Conf. (COMPSAC)*, Kyoto, Japan, 2013, pp. 754–759.

[21] Y.-F. Lu and S.-S. Ye, "A multi-dimension hash index design for main-memory RFID database applications," in *Proc. Int. Conf. Inf. Security Intell. Control (ISIC)*, 2012, pp. 61–64.

[22] T. Hara, K. Harumoto, M. Tsukamoto, S. Nishio, and J. Okui, "Main memory database for supporting database migration," in *Proc. IEEE Pac. Rim Conf. Commun. Comput. Signal Process. 10 Years PACRIM 1987–1997 Netw. Pac. Rim*, vol. 1. Victoria, BC, Canada, 1997, pp. 231–234.

[23] N. Martinez-Bazan, S. Gomez-Villamor, and F. Escale-Claveras, "DEX: A high-performance graph database management system," in *Proc. IEEE 27th Int. Conf. Data Eng. Workshops (ICDEW)*, Hanover, Germany, 2011, pp. 124–127.

[24] D. Brickley and R. V. Guha, *RDF Vocabulary Description Language 1.0: RDF Schema*, 2004. [Online]. Available: http://www.w3.org/2001/sw/RDFCore/Schema/200203

[25] Z. Kaoudi and I. Manolescu, "Cloud-based RDF data management," in *Proc. ACM SIGMOD Int. Conf. Manag. Data*, 2014, pp. 725–729.

[26] M. Grund, M. Schapranow, J. Krueger, J. Schaffner, and A. Bog, "Shared table access pattern analysis for multi-tenant applications," in *Proc. IEEE Symp. Adv. Manag. Inf. Glob. Enterprises (AMIGE)*, Tianjin, China, 2008, pp. 1–5.

[27] H. Cai *et al.*, "IoT-based configurable information service platform for product lifecycle management," *IEEE Trans. Ind. Informat.*, vol. 10, no. 2, pp. 1558–1567, May 2014.

[28] J. Espadas *et al.*, "A tenant-based resource allocation model for scaling software-as-a-service applications over cloud computing infrastructures," *Future Gener. Comput. Syst.*, vol. 29, no. 1, pp. 273–286, 2013.

[29] A. K. Singh, X. Cui, B. Cassell, B. Wong, and K. Daudjee, "MicroFuge: A middleware approach to providing performance isolation in cloud storage systems," in *Proc. IEEE 34th Int. Conf. Distrib. Comput. Syst. (ICDCS)*, Madrid, Spain, 2014, pp. 503–513.

[30] P. Lama and X. Zhou, "NINEPIN: Non-invasive and energy efficient performance isolation in virtualized servers," in *Proc. IEEE/IFIP Int. Conf. Depend. Syst. Netw. (DSN)*, Boston, MA, USA, 2012, pp. 1–12.

[31] B. Li, Y. He, and K. Xu, "Distributed metadata management scheme in cloud computing," in *Proc. 6th Int. Conf. Pervasive Comput. Appl. (ICPCA)*, Port Elizabeth, South Africa, 2011, pp. 32–38.

[32] Y. Wang and H. Lv, "Efficient metadata management in cloud computing," in *Proc. IEEE 3rd Int. Conf. Commun. Softw. Netw. (ICCSN)*, Xi'an, China, 2011, pp. 514–519.

[33] R. Anitha and S. Mukherjee, *Global Trends in Information Systems and Software Applications* (Communications in Computer and Information Science). Heidelberg, Germany: Springer, 2012, pp. 13–21.

[34] M. A. Rodriguez, J. Bollen, and H. V. D. Sompel, "Automatic metadata generation using associative networks," *ACM Trans. Inf. Syst.*, vol. 27, no. 2, 2009, Art. no. 7.

[35] M. Á. Rodríguez-García, R. Valencia-García, F. García-Sánchez, and J. J. Samper-Zapater, "Ontology-based annotation and retrieval of services in the cloud," *Knowl. Based Syst.*, vol. 56, pp. 15–25, Jan. 2014.

[36] J. C. Vidal, M. Lama, E. Otero-García, and A. Bugarín, "Graph-based semantic annotation for enriching educational content with linked data," *Knowl. Based Syst.*, vol. 55, pp. 29–42, Jan. 2014.

[37] D. Tao, L. Jin, W. Liu, and X. Li, "Hessian regularized support vector machines for mobile image annotation on the cloud," *IEEE Trans. Multimedia*, vol. 15, no. 4, pp. 833–844, Jun. 2013.

[38] L. Xu and X. Wu, "Hub: Heterogeneous bucketization for database outsourcing," in *Proc. Int. Workshop Security Cloud Comput.*, Wuhan, China, 2013, pp. 47–54.

[39] Y. Ma *et al.*, "An efficient index for massive IoT data in cloud environment," in *Proc. 21st ACM Int. Conf. Inf. Knowl. Manag.*, Maui, HI, USA, 2012, pp. 2129–2133.

[40] J. Chen *et al.*, "Walnut: A unified cloud object store," in *Proc. ACM SIGMOD Int. Conf. Manag. Data*, Scottsdale, AZ, USA, 2012, pp. 743–754.

[41] S. Huston, J. S. Culpepper, and W. B. Croft, "Indexing word sequences for ranked retrieval," *ACM Trans. Inf. Syst.*, vol. 32, no. 1, 2014, Art. no. 3.

[42] S. Kamara, C. Papamanthou, and T. Roeder, "Dynamic searchable symmetric encryption," in *Proc. ACM Conf. Comput. Commun. Security*, Hangzhou, China, 2012, pp. 965–976.

[43] S. K. Vishwakarma, K. I. Lakhtaria, D. Bhatnagar, and A. K. Sharma, "An efficient approach for inverted index pruning based on document relevance," in *Proc. 4th Int. Conf. Commun. Syst. Netw. Technol. (CSNT)*, Bhopal, India, 2014, pp. 487–490.

[44] S. Blanas *et al.*, "A comparison of join algorithms for log processing in MapReduce," in *Proc. ACM SIGMOD Int. Conf. Manag. Data*, Indianapolis, IN, USA, 2010, pp. 975–986.

[45] K. Lu *et al.*, "Wave: Trigger based synchronous data process system," in *Proc. 14th IEEE/ACM Int. Symp. Cluster Cloud Grid Comput. (CCGrid)*, Chicago, IL, USA, 2014, pp. 540–541.

[46] A. Lugowski *et al.*, "Parallel processing of filtered queries in attributed semantic graphs," *J. Parallel Distrib. Comput.*, vols. 79–80, pp. 115–131, May 2015.

[47] K. Lee, L. Liu, Y. Tang, Q. Zhang, and Y. Zhou, "Efficient and customizable data partitioning framework for distributed big RDF data processing in the cloud," in *Proc. IEEE CLOUD*, Santa Clara, CA, USA, 2013, pp. 327–334.

[48] F. N. Afrati and J. D. Ullman, "Optimizing multiway joins in a map-reduce environment," *IEEE Trans. Knowl. Data Eng.*, vol. 23, no. 9, pp. 1282–1298, Sep. 2011.

[49] G. Jung, N. Gnanasambandam, and T. Mukherjee, "Synchronous parallel processing of big-data analytics services to optimize performance in federated clouds," in *Proc. IEEE 5th Int. Conf. Cloud Comput. (CLOUD)*, Honolulu, HI, USA, 2012, pp. 811–818.

[50] M. Spreitzer, M. Steinder, and I. Whalley, "Ripple: Improved architecture and programming model for bulk synchronous parallel style of analytics," in *Proc. IEEE 33rd Int. Conf. Distrib. Comput. Syst. (ICDCS)*, Philadelphia, PA, USA, 2013, pp. 460–469.

[51] D. B. Babu, R. S. R. Prasad, and N. K. Chakravarthy, "A modern cyclic approach to solve a classification problem in cloud environment," in *Proc. Int. Conf. Adv. Comput. Sci. Appl. Technol. (ACSAT)*, Kuching, Malaysia, 2013, pp. 207–212.

[52] X. Zhengqiao and Z. Dewei, "Research on clustering algorithm for massive data based on Hadoop platform," in *Proc. Int. Conf. Comput. Sci. Service Syst. (CSSS)*, Nanjing, China, 2012, pp. 43–45.

[53] J. Yu and T. Zhu, "Towards dynamic resource provisioning for traffic mining service cloud," in *Proc. IEEE Internet Things (iThings/CPSCom) IEEE Int. Conf. IEEE Cyber Phys. Soc. Comput. Green Comput. Commun. (GreenCom)*, Beijing, China, 2013, pp. 1296–1301.

[54] S. Pandey, N. Gupta, and A. K. Dubey, "A novel wireless heterogeneous data mining (WHDM) environment based on mobile computing environments," in *Proc. Int. Conf. Commun. Syst. Netw. Technol. (CSNT)*, 2011, pp. 298–302.

[55] F. T. Stahl, M. M. Gaber, M. Bramer, and P. S. Yu, "Distributed hoeffding trees for pocket data mining," in *Proc. Int. Conf. High Perform. Comput. Simulat. (HPCS)*, Istanbul, Turkey, 2011, pp. 686–692.

[56] E. H.-C. Lu, W.-C. Lee, and V. S.-M. Tseng, "A framework for personal mobile commerce pattern mining and prediction," *IEEE Trans. Knowl. Data Eng.*, vol. 24, no. 5, pp. 769–782, May 2012.

[57] C.-C. Chen, K.-W. Lee, C.-C. Chang, D.-N. Yang, and M.-S. Chen, "Efficient large graph pattern mining for big data in the cloud," in *Proc. IEEE Int. Conf. Big Data*, Silicon Valley, CA, USA, 2013, pp. 531–536.

[58] H.-C. Lai, C.-T. Li, Y.-C. Lo, and S.-D. Lin, "Exploiting and evaluating MapReduce for large-scale graph mining," in *Proc. IEEE/ACM Int. Conf. Adv. Soc. Netw. Anal. Min. (ASONAM)*, Istanbul, Turkey, 2012, pp. 434–441.

[59] J. Wu, Z. Guan, Q. Zhang, A. K. Singh, and X. Yan, "Static and dynamic structural correlations in graphs," *IEEE Trans. Knowl. Data Eng.*, vol. 25, no. 9, pp. 2147–2160, Sep. 2013.

[60] Y. Lyu *et al.*, "High-performance scheduling model for multisensor gateway of cloud sensor system-based smart-living," *Inf. Fusion*, vol. 21, pp. 42–56, Jan. 2015.

[61] Z. Peng, Z. Jingling, and L. Qing, "Message oriented middleware data processing model in Internet of Things," in *Proc. 2nd Int. Conf. Comput. Sci. Netw. Technol. (ICCSNT)*, Changchun, China, 2012, pp. 94–97.

[62] F. Nordemann, "A communication-optimizing middleware for efficient wireless communication in rural environments," in *Proc. 9th Middleware Doctoral Symp. 13th ACM/IFIP/USENIX Int. Middleware Conf.*, Montreal, QC, Canada, 2012, p. 3.

[63] Y. Wang, Q. Deng, W. Liu, and B. Song, "A data-centric storage approach for efficient query of large-scale smart grid," in *Proc. 9th Web Inf. Syst. Appl. Conf. (WISA)*, Haikou, China, 2012, pp. 193–197.
[64] Y. Tan, Z. Yan, D. Feng, E. H.-M. Sha, and X. Ge, "Reducing the delinearization of data placement to improve deduplication performance," in *Proc. High Perform. Comput. Netw. Stor. Anal. (SCC) SC Companion*, Salt Lake City, UT, USA, 2012, pp. 796–800.
[65] B. Dong *et al.*, "An optimized approach for storing and accessing small files on cloud storage," *J. Netw. Comput. Appl.*, vol. 35, no. 6, pp. 1847–1862, 2012.
[66] F. Ning, C. Weng, and Y. Luo, "Virtualization I/O optimization based on shared memory," in *Proc. IEEE Int. Conf. Big Data*, Silicon Valley, CA, USA, 2013, pp. 70–77.
[67] L. Kolb, A. Thor, and E. Rahm, "Load balancing for MapReduce-based entity resolution," in *Proc. IEEE 28th Int. Conf. Data Eng.*, Washington, DC, USA, 2012, pp. 618–629.
[68] F. Marozzo, D. Talia, and P. Trunfio, "P2P-MapReduce: Parallel data processing in dynamic cloud environments," *J. Comput. Syst. Sci.*, vol. 78, no. 5, pp. 1382–1402, 2012.
[69] P. Lu, Y. C. Lee, and A. Y. Zomaya, "Non-intrusive slot layering in Hadoop," in *Proc. 13th IEEE/ACM Int. Symp. Cluster Cloud Grid Comput. (CCGrid)*, Delft, The Netherlands, 2013, pp. 253–260.
[70] J. Zhou *et al.*, "CloudThings: A common architecture for integrating the Internet of Things with cloud computing," in *Proc. IEEE 17th Int. Conf. Comput. Supported Cooperative Work Design (CSCWD)*, Whistler, BC, Canada, 2013, pp. 651–657.
[71] K. Tei and L. Gürgen, "ClouT: Cloud of things for empowering the citizen clout in smart cities," in *Proc. IEEE World Forum Internet Things (WF-IoT)*, Seoul, South Korea, 2014, pp. 369–370.
[72] L. Jiang *et al.*, "An IoT-oriented data storage framework in cloud computing platform," *IEEE Trans. Ind. Informat.*, vol. 10, no. 2, pp. 1443–1451, May 2014.
[73] J. Byun, S. H. Kim, and D. Kim, "Lilliput: Ontology-based platform for IoT social networks," in *Proc. IEEE Int. Conf. Services Comput. (SCC)*, Anchorage, AK, USA, 2014, pp. 139–146.
[74] M. Fiorelli, M. T. Pazienza, A. Stellato, and A. Turbati, "ART lab infrastructure for semantic big data processing," in *Proc. Int. Conf. High Perform. Comput. Simulat. (HPCS)*, Bologna, Italy, 2014, pp. 327–334.
[75] C. Bizer, T. Heath, and T. Berners-Lee, "Linked data-the story so far," *Int. J. Semantic Web Inf. Syst.*, vol. 5, no. 3, pp. 1–22, 2009.
[76] R. Haque and M.-S. Hacid, "Blinked data: Concepts, characteristics, and challenge," in *Proc. IEEE World Congr. Services*, Anchorage, AK, USA, 2014, pp. 426–433.
[77] C. Xie, G. Li, H. Cai, L. Jiang, and N. N. Xiong, "Dynamic weight-based individual similarity calculation for information searching in social computing," *IEEE Syst. J.*, to be published, doi: 10.1109/JSYST.2015.2443806.
[78] T. Michelsen, "Data stream processing in dynamic and decentralized peer-to-peer networks," in *Proc. SIGMOD PhD Symp.*, 2014, pp. 1–5.
[79] S. Aridhi, L. d'Orazio, M. Maddouri, and E. M. Nguifo, "Density-based data partitioning strategy to approximate large-scale subgraph mining," *Inf. Syst.*, vol. 48, pp. 213–223, Mar. 2015.
[80] C. Doulkeridis and K. Nørvåg, "A survey of large-scale analytical query processing in MapReduce," *VLDB J.*, vol. 23, no. 3, pp. 355–380, 2014.

Hongming Cai (M'10–SM'15) received the B.S., M.S., and Ph.D. degrees from Northwestern Polytechnical University, Xi'an, China, in 1996, 1999, and 2002, respectively.

He is currently an Associate Professor with the School of Software, Shanghai Jiao Tong University. He served as a Post-Doctoral Research Fellow with the Computer Science and Technology Department, Shanghai Jiao Tong University, Shanghai, China, from 2002 to 2004. He served as a Visiting Professor with the Business Information Technology Institute, University of Mannheim, Mannheim, Germany, from 2008 to 2009. His visiting scholarship was appointed and sponsored by Alfried Krupp von Bohlen und Halbach Foundation, Germany.

Dr. Cai was a recipient of the National Outstanding Scientific and Technological Workersİ by China Association for Science and Technology in 2012. He is the Standing Director of China Graphics Society and a Senior Member of the ACM.

Boyi Xu (M'14) received the B.S. degree in industrial automation and Ph.D. degree in management science from Tianjin University, Tianjin, China, in 1987 and 1996, respectively.

He is currently an Associate Professor with the College of Economics and Management, Shanghai Jiao Tong University, Shanghai, China. His current research interests include enterprise information systems, Internet of Things, and business intelligence.

Lihong Jiang (M'10) received the B.S., M.S., and Ph.D. degrees from Tianjin University, Tianjin, China, in 1989, 1992, and 1996, respectively.

From 1992 to 1993, she was as an Assistant Professor with the Department of Computer, Qingdao Ocean University, Qingdao, China. From 1996 to 1998, she was as a Post-Doctoral Research Fellow with the School of Management, Fudan University, Shanghai, China. She is currently an Associate Professor with the School of Software, Shanghai JiaoTong University, Shanghai.

Athanasios V. Vasilakos (M'00–SM'11) is currently a Professor with the Luleå University of Technology, Luleå, Sweden.

Prof. Vasilakos has served or is serving as an Editor for many technical journals, such as the IEEE TRANSACTIONS ON NETWORK AND SERVICE MANAGEMENT, the IEEE TRANSACTIONS ON CLOUD COMPUTING, the IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY, the IEEE TRANSACTIONS ON CYBERNETICS, the IEEE TRANSACTIONS ON NANOBIOSCIENCE, the IEEE TRANSACTIONS ON INFORMATION TECHNOLOGY IN BIOMEDICINE, *ACM Transactions on Autonomous and Adaptive Systems*, and the IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS. He is also the General Chair of the European Alliances for Innovation.