IEEE Access
Multidisciplinary ⁝ Rapid Review ⁝ Open Access Journal

Received June 6, 2018, accepted July 14, 2018, date of publication July 20, 2018, date of current version August 15, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2857845

VPL-Based Big Data Analysis System: UDAS

HYUNJIN CHOI[1], JANGWON GIM[2], YOUNG-DUK SEO[3], AND DOO-KWON BAIK[3]
[1]Department of Computer and Radio Communications Engineering, Korea University, Seoul 02841, South Korea
[2]Department of Software Convergence Engineering, Kunsan National University, Gunsan 54150, South Korea
[3]Department of Computer Science and Engineering, Korea University, Seoul 02841, South Korea

Corresponding authors: Jangwon Gim (jwgim@kunsan.ac.kr) and Doo-Kwon Baik (baikdk@korea.ac.kr)

This work was supported in part by the Institute for Information and communications Technology Promotion (IITP) grant funded by the Korean Government (MSIP) (big data-based unified data analytics suite in the cloud) under Grant 2015-0-00521 and in part by the National Research Foundation of Korea (NRF) grant funded by the Korean Government (MSIT) under Grant NRF-2018R1C1B6008624.

ABSTRACT Over the past five years, research on big data analysis has been actively conducted, and many services have been developed to find valuable data. However, low quality of raw data and data loss problem during data analysis make it difficult to perform accurate data analysis. With the enormous generation of both unstructured and structured data, refinement of data is becoming increasingly difficult. As a result, data refinement plays an important role in data analysis. In addition, as part of efforts to ensure research reproducibility, the importance of reuse of researcher data and research methods is increasing; however, the research on systems supporting such roles has not been conducted sufficiently. Therefore, in this paper, we propose a big data analysis system named the unified data analytics suite (UDAS) that focuses on data refinement. UDAS performs data refinement based on the big data platform and ensures the reusability and reproducibility of refinement and analysis through the visual programming language interface. It also recommends open source and visualization libraries to users for statistical analysis. The qualitative evaluation of UDAS using the functional evaluation factor of the big data analysis platform demonstrated that the average satisfaction of the users is significantly high.

INDEX TERMS Data analysis, data visualization, reproducibility of results, clouds, data refinement, R.

I. INTRODUCTION

Recently, the amount of data has been increased exponentially due to the dispersion of data from social network services (SNSs), internet of things (IoT), and cloud services [1]–[4]. These huge amounts of data are called as big data [5]. Big data itself is not significant, but we can create significant value by refining and analyzing it. Therefore, many studies on big data analysis have been conducted in various research fields and many software and systems have been developed to assist in analyzing structured and unstructured data; such as, R, SAS, and SPSS [6]–[8]. There are three main processes for data analysis in the business field [9]: data collection, data refinement, and data delivery. First, the data collection process aims to collect data quickly and reliably. Data is collected in real time or batch according to the purpose of analysis and the type of service to be provided. Recently, open source tools such as Amazon Kinesis and Apache Kafka have been developed for real-time data collection, enabling stable and fast data collection [10]–[12]. The data refinement process cleans up the collected data by determining which data should be integrated according to the analysis requirements and purposes, and then integrates the refined data. This data refinement process plays an important role in the statistical analysis of big data for improving the accuracy and reliability of visualization services [13]. Finally, the data delivery process communicates the results of the analysis to the user through statistical analysis or data visualization, and also allows data scientists to share insights regarding analysis results [6], [14].

Despite of the importance of data refinement, this is costly and time-consuming process [15], [16]. Depending on the purpose and method of analysis, existing scripts and programs are frequently modified for data refinement. Therefore, it is necessary to verify not only the data refinement process itself but also the results from this process, because data may be lost owing to execution error when the script is executed manually during the refinement process, which is performed routinely and repeatedly. All these processes must be monitored and managed by data scientists; however, it is not easy to share these data refinement procedures and knowledge when the analytical data and domains are changed. Therefore, to reduce the cost and time required for data analysis, we need a solution for efficient and convenient data refinement process [17]. Recently, research related to data collection and data delivery have been actively conducted, and various tools and systems have been developed in the big data field; however, research regarding tools focused on data refinement are insufficient [18]–[20]. OpenRefine [21] is widely used as a tool for data refinement. This tool can expand the knowledge graph substantially by supporting data wrangling functions such as data syntax checking, data clustering, and data cleansing. However, the detailed procedures and methods for the data refinement process are not provided. Therefore, when integrating knowledge graphs for new raw data, much time and effort is required for the data scientist to perform iterative data wrangling process.

In this paper, to reduce the cost of existing approaches for data analysis, we define methods for data refinement and implement a data analysis system focusing on data refinement to consider the convenience of users. In other words, we propose a big data refinement and statistical analysis system based on the visual programming language (VPL) concept [22].

The main contributions of this paper are as follows:

- Big data collection, refinement, statistical analysis, and visualization system in the cloud environment: Capable of collecting unstructured text data and structured data, refining and analyzing through the cloud environment, and providing a visualization service of the analysis results on the web interface.
- Ensure reusability of the data refinement process: Data refinement process design, data wrangling, data mapping, and data integration can all be performed via the VPL interface, depending on the data analysis requirements. The data refinement process can be configured using the drag and drop method, and the process can be saved and shared as a template so that users can perform the refinement process easily.
- Ensure reproducibility of open source-based data analysis: R packages and functions widely used for statistical analysis can be programmed through the VPL interface, and the generated data analysis model can be stored and used by users such as domain experts, and the executed process can be reproduced. In other words, the proposed system ensures reproducibility of research methodology. In addition, various visualization results can be obtained by applying the visualization library provided in D3.js through the proposed system [23].

The remainder of this paper is organized as follows. In section 2, we describe the existing data analysis tools, particularly focusing on data refinement function, and VPL-based data analysis tools. Section 3 explains the UDAS architecture in detail. Section 4 describes an implementation of the UDAS, focusing on the main modules. In section 5, we evaluate the UDAS on the basis of a use case study and qualitative evaluation. Finally, section 6 presents a brief conclusion.

II. RELATED WORK

A. DATA ANALYSIS TOOLS

As the importance of data analysis has increased in big data environments, software tools supporting data collection, data refinement, and data delivery for data analysis, such as R [24], SAS [25], and SPSS [26], have been developed. Most tools provide a data delivery function focusing on statistical analysis and data visualization; however, there are no tools that cover three major functions for data analysis, with the exception of Deducer [27] and SAS. In addition, limited works have been conducted on the provision of the data refinement function, which is the most important function in the data analysis process. Table 1 represents the classification of existing data analysis tools. They are classified into three main categories: functionality, feature, and environment. Functionality represents the main three processes of data analysis. The feature category represents the ability of tools to focus on reproducibility and reusability. Finally, in environment category, we examine the availabilities of cloud services and open source software for data analysis.

Especially, data analysis tools are predominantly based on R, which is a free programming language for statistical computing. Open source R has recently been widely used in various fields for big data analysis. Researchers can freely customize the published R libraries for research purposes and apply the various analysis models easily. As a result, the use of R has increased in recent years not only in the analysis of big data, but also in visualization of results of machine learning. Various tools and interfaces related to R have been developed, such as RStudio [28], RKWard [29], JGR [30], RCommander [31], Rattle [32], and Deducer. RStudio is the most well-known and representative open source data analysis software for R. It is the most popular of all data analysis tools related to R. RKWard, JGR, and R Commander have been developed to facilitate easier use and higher efficiency of the R language and R packages to focus on the GUI. Recently, tools for various operating system environments have been developed to guarantee platform independency and enable analysis in the big data ecosystem; these tools include Rattle and Deducer.

Nevertheless, the R engine has structural limitations that prevent satisfactory performance in analyzing large amounts of data. To solve these problems and to ensure the above-mentioned advantages of using R, there are projects that consider distributed and cloud computing. Apache Spark has released SparkR [33] for interoperability with R. SparkR supports the interoperability between Spark data frames and R data frames, which are capable of memory-based distributed processing, and it used in large-scale data analysis and visualization. Microsoft has released the Microsoft Machine Learning Server [34]–[36], an open source analytics platform. Users can conduct analysis models using R libraries in a cloud through Microsoft Machine Learning Server. In this way, projects related to the R engine are implemented in the distributed processing and cloud computing for big data

TABLE 1. **Classification of existing data analysis tools.**

Tools	Functionality			Feature		Environment	
	Collection	Refinement	Delivery	Reproducibility	Reusability	Cloud Service	Open Source
RStudio [29]	Weakly supports	n/a	Supports	n/a	Weakly supports	n/a	Supports
RKWard [30]	Weakly supports	n/a	Supports	n/a	Weakly supports	n/a	Supports
JGR [31]	Weakly supports	n/a	Supports	n/a	Weakly supports	n/a	Supports
R Commander [32]	Weakly supports	n/a	Supports	n/a	Weakly supports	n/a	Weakly supports
Rattle [33]	n/a	n/a	Supports	n/a	Weakly supports	n/a	Supports
Deducer [28]	Supports	Weakly support	Supports	n/a	Weakly supports	n/a	n/a
SAS [26]	Supports	Weakly supports	Supports	n/a	Weakly supports	n/a	n/a
SPSS [27]	Weakly supports	n/a	Supports	n/a	Weakly supports	n/a	n/a
SRC-STAT [38]	n/a	Weakly supports	Supports	n/a	n/a	n/a	Supports
Stat! [39]	n/a	Weakly supports	Weakly supports	n/a	n/a	n/a	n/a
ManyEyes [40]	Weakly supports	n/a	Supports	n/a	Weakly supports	n/a	Supports
OpenRefine [22]	n/a	Supports	n/a	n/a	Weakly supports	n/a	Supports
UDAS	Supports	Supports	Supports	Supports	Supports	Supports	Supports

processing, and the use of R is increasing to enhance the productivity of the researchers and to support the reproducibility of the research methods.

Most R-based data analysis tools focus on the data delivery, because R is a strong tool for statistical analysis and data visualization. However, there is no tool to support all the data analysis processes, and in particular, they do not offer the data refinement process. Deducer provides a data refinement function, but it is not fully functional.

SAS and SPSS are the most representative statistical analysis tools similar to R. There are some differences between SPSS and SAS; SPSS is a statistical analysis tool that focuses only on the statistical analysis algorithm, whereas SAS is able to provide various data analysis functions, as well as a statistical analysis algorithm. They both provide powerful data analysis functions in terms of the data delivery, and partially support the data collection. However, considerable time and costs are required for users to use them, because they were both developed for experts, such as statisticians and data scientists. SRC-STAT [37] was developed considering the convenience of users who are unfamiliar with statistical processing and analysis. This tool adds visual analysis functions, such as a scatter plot matrix explorer, hierarchical clustering explorer, and data visualization explorer, to provide a GUI for users to explore and analyze data easily. SRC-STAT has a scatter plot matrix explorer and a hierarchical clustering explorer. Based on these functions SRC-STAT can refine data. However, users must possess substantial statistical knowledge to use SRC-STAT; besides, exploring the data intuitively is difficult. Furthermore, SRC-STAT is not suitable for the big data environment because it does not consider the processing of big data. Stat! [38] refines appropriate queries for data scientists to explore data and provides tables and graphs of data analysis results through multiple queries. However, Stat! does not provide the ultimate purpose of the data refinement function, because it only refines the query for the data explorer. In addition, it supports some tables and graphs, but not various data visualization functions, and the UI is not user-friendly. ManyEyes [39] is a web-based system that allows users to collaborate on data analysis and visualization. Users can obtain visualization results and the visualization results can be modified via feedback from other users using web-based interface. Communication among users is the most important requirement of this web-based system. However, without active communication, achieving the goal of this system is difficult. In addition, this system does not support the data collection and refinement functions.

OpenRefine is the only data analysis tool that focuses fully on the data refinement function. This tool provides fundamental refinement functions, such as data cleaning and data transformation, as well as automation for refinement, which allows users to reduce the time and cost required for data refinement. Furthermore, it is useful for identifying trends in big data sets, and allows expanding the data by linking with a knowledge base such as Wikipedia. However, OpenRefine does not store the refinement process worked on by the users. Therefore, if users conduct the same data analysis process again, the cost of data refinement is incurred as in any other tools. In other words, OpenRefine does not support reusability and reproducibility. In existing data analysis tools, the data analysis processes have not been developed to enable linking data refinement and visualization as a series of processes, but rather as independent modules. Therefore, there is a difficulty in integrating and linking the results of each process. Furthermore, the linkage and integration information of data for the continuous processes is missing; thus, loss of implicitly derivable information can occur.

The proposed UDAS provides all functions necessary for data analysis, such as data collection, refinement, and delivery. We establish the process for data refinement in the UDAS to reduce the cost of refinement. In addition, the UDAS stores the continuous processes from data refinement to visualization as a template. To use this template, users can perform the processes necessary for data analysis easily compared with other tools. The UDAS system also ensures the advantages of open source R in the cloud computing and uses the R library flexibly. The VPL-based interface of UDAS facilitates data collection, processing, and refinement, as well as data analysis in distributed and cloud computing.

B. VISUAL PROGRAMMING LANGUAGE (VPL)

A programming language that uses a visual representation such as graphics, drawings, animation, or icons, partially or completely is named VPL. It can help manipulate visual information, support visual interaction, and allow programming with visual expressions. VPL is a set of spatial arrangements of text/graphic symbols with a semantic interpretation that is used in generating communication data. VPL has several advantages such as fewer programming concepts, immediate visual feedback, and explicit depiction of relationships. Thus, many tools have been developed to analyze big data using the VPL concept, such as Azure [40], Orange [41], and KNIME [42].

Azure is a representative platform that applies VPL and it was designed based on a graphical dataflow-based programming model. Azure is targeted for beginner programmers and users, with a basic understanding of tools and functions, who want to analyze big data without the help of data scientists or statistical experts. Therefore, Azure can be used to conduct rapid prototyping or code development to analyze big data. In addition to Azure, Orange and KNIME feature a user-friendly programming language based on VPL, making it easier for users to integrate data.

Existing VPL-based tools and platforms focus on the ease of use of the programming language for users. However, they do not consider the various and complex situations, such as BI and BA. Furthermore, there are little works for big data analysis tools based on VPL. IBM Cognos [43] is a representative system based on VPL for data analysis, and it supports actively flexible programming to take advantage of VPL for diversifying business situations. Nonetheless, Cognos does not support important functions for data collection and refinement in big data analysis. However, the UDAS system proposed in this paper provides users with three most important functions for analyzing big data such as data collection, data refinement, and data delivery.

III. SYSTEM ARCHITECTURE

The UDAS proposed in this paper supports the design and execution of the refining and analysis process by applying VPL concepts, including data collection, refinement, and analysis functions, for big data analysis. Fig. 1 shows the overall system architecture. The UDAS consists of three stages: data collection, data refinement, and data analysis. In the first step, the data collection phase, various types of big data (e-mail, SNS, Web documents, RDB, Web log files, etc.) are collected in real time and in batch mode using each collection agent. In addition, the collected raw data is referred to the data analysis policy, and data schema stored in the Data Orchestrator for decomposition and reintegration according to the purpose of analysis and data type. A message queuing module can control and save each data in a Hadoop cluster capable of distributed parallel processing, as shown in Fig. 2.

The second step, data refinement, includes the data manipulation function that includes the data wrangler function, data query, and data mapping necessary to design the logical and physical model of the data to analyze and refine the data. As a result, a refinement process can be created and stored as a template. At this time, the target data used in the refining process, refinement table information, mapping rules, and the order of the refinement process are stored in the UDAS as metadata to ensure reusability and reproducibility of the template. Therefore, non-specialists as well as domain experts can reuse or customize the data refinement process by selecting the template. This is because the UDAS supports the entire process from data collection to refinement and analysis to be modularized procedurally, and executed step by step. The main components of the UDAS are as follows.

A. ER DESIGNER

ER designer is a component used for designing accurate data models (logical and physical) for data analysis. It supports data duplication prevention and data normalization.

B. QUERY DESIGNER

Query designer supports accurate and rapid generation of various queries (inner join, outer join, cross join, and other joins) through a VPL-based query design interface.

C. MAPPING DESIGNER

Data wrangling is the task of manipulating raw data manually for data analysis or integrating modified data with other interoperable data. As a result, data wrangling work is made up of various scripts used in R, and this work is difficult to reuse existing scripts for new incoming raw data without changing the purpose of analysis. Through the VPL-based GUI interface, the refined data are mapped to the physical schema of the refinement table while diagnosing and processing the data of the wrangling object.

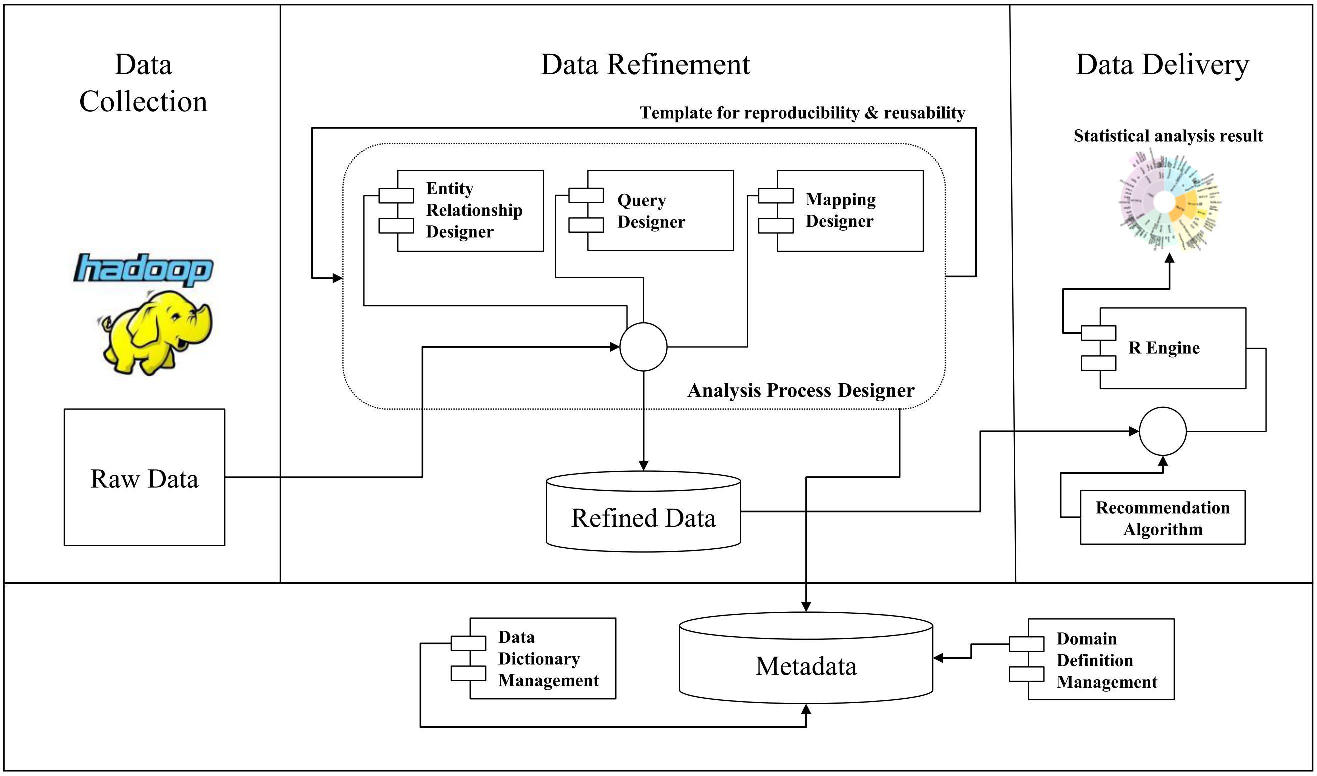

Data Collection
Data Refinement
Data Delivery
Template for reproducibility & reusability
Statistical analysis result
Entity Relationship Designer
Query Designer
Mapping Designer
hadoop
Analysis Process Designer
R Engine
Raw Data
Refined Data
Recommendation Algorithm
Data Dictionary Management
Metadata
Domain Definition Management

FIGURE 1. Architecture of UDAS.

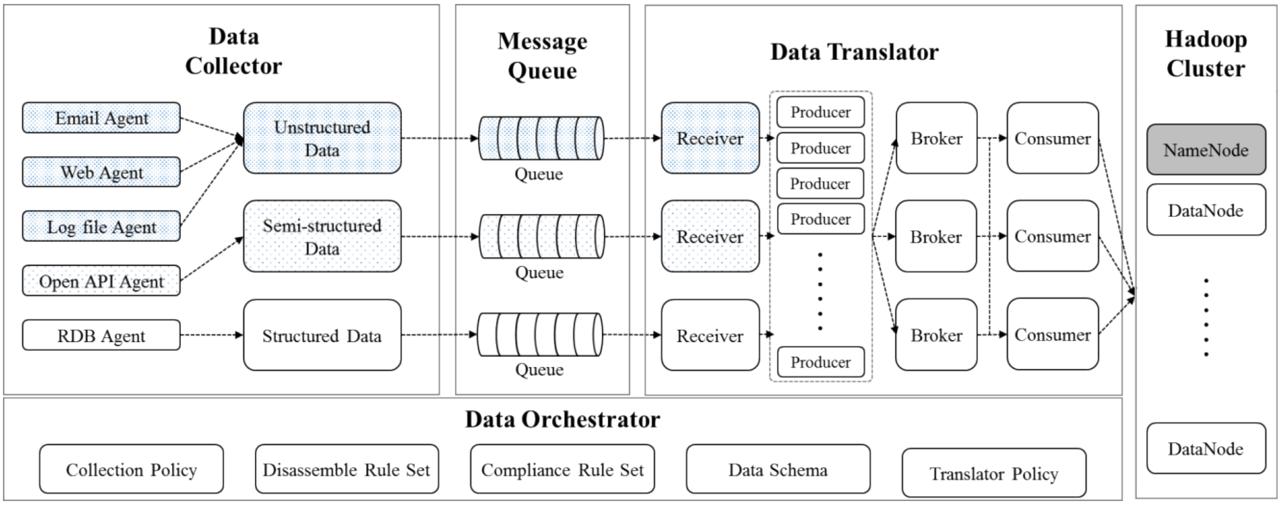


FIGURE 2. Data collection process in the UDAS.

D. DATA DICTIONARY

To ensure the interoperability of the terms used in the data refining and analysis process, a data dictionary is created for ensuring the consistency of terms. Specifically, the data dictionary is used to unify the terms used in the schema for the analysis data and the representative terms in the processing of the original data. As a result, it is possible not only to prevent lexical inconsistency in analyzing data generated in the domain field (financial, sensor data, log analysis, etc.), but also to manage standard terms by controlling terms that can be used differently by user change.

E. DATA DOMAIN COMPONENT

To ensure the accuracy of data analysis, it is necessary to manage primitive types of data (e.g., integer, float, double, etc.). In the process of generating analysis data, some raw data may be lost owing to data type conversion (e.g., float to double). Therefore, the data quality should be managed throughout the data analysis. To this end, the data domain management component of the UDAS refers to the data schema for analysis and performs continuous management of the domain of data.

F. TEMPLATE FOR REFINEMENT AND ANALYSIS

An analytical project generated through the UDAS includes at least one or more refining processes and analytical models. The refinement processes and analysis models used in project can be saved as a template after completion of the project. Therefore, user can reuse this templet next time, and it means that UDAS guarantee reusability.

G. STATISTICAL ANALYSIS AND VISUALIZATION

The analytical model of the UDAS recommends a statistical analysis library applicable to the data considering the types of data (e.g., nominal, discrete, etc.).

The analytical model uses a library of open source R, which is widely used in statistical analysis. To achieve this, we define the metadata by analyzing the representative analytic functions defined in R (2,583 functions in total). The UDAS recommends functions applicable to the analysis model through defined metadata and the analysis of data type. The UDAS also applies D3.js, which includes various libraries based on JavaScript, for visualization of the analysis results. To achieve this, we analyze 150 graphs used in D3.js and define the metadata. As a result, the UDAS supports various visualizations of data frames generated from the analysis model.

IV. IMPLEMENTATION

This chapter discusses the implementation of the UDAS defined in Chapter 3.

A. DATA COLLECTION

The data collection step performs the function of collecting various types of data (structured, unstructured, and semi-structured) in real time. For this purpose, the UDAS is implemented based on the Hadoop ecosystem for storing and processing large amounts of unstructured text data (e.g., e-mail, log file, text web document, etc.). In addition, it supports relational databases and machine-readable data such as CSV and XML. The UDAS can monitor the status of the data collection step from the relational database in the dashboard, along with unstructured data such as e-mail and web logs. Further, it is possible to monitor the occurrence of errors during collection and cope with the data collection stage. In addition, the vast amount of data collected in real time is stored according to subject classification and data characteristics by the purpose of analysis using Apache Kafka and a message queuing system.

B. DATA REFINEMENT

First, in the data refinement step, UDAS performs the data extraction and integration process for analysis to reproduce raw data as analyzed data. In these processes, the user designs the table schema to select the data to be analyzed. Then, data mapping is performed based on the designed table schema, where the raw data are reproduced and integrated into clean data by data wrangling process. The data type definitions and data mapping rules which are used during data refinement are stored with the order information of the refinement process. As a result, the refinement process is reusable for analytical models that require the same refinement process. In addition, a single refinement process can be distinguished into several sub steps, which can be applied to new analytical models by disassembling and reassembling the refinement process for the purpose of analysis. The UDAS can visualize the data refining process through the VPL-based GUI interface to grasp the flow of the refinement process so that it can change the configuration of the refinement process and assembling the lower-level component. In addition, the refinement of unstructured data is developed in different types of scripts or programming languages, taking into consideration the type of data and the purpose of analysis. Therefore, it is not easy to reuse individual scripts or programs. Thus, this supports the reusability of existing scripts and libraries in the refinement model and the analysis model. Fig. 3 shows the IDE screen for manipulating and executing the data refinement process. The interface of data refinement process includes the following models.

1) REFINEMENT MODEL

The refinement process consists of several individual components, and each component produces the next step of input data according to the data refinement flow. In other words, the refinement component represents the data to be refined and the wrangling operation on that data. Q notation in Fig. 3 indicates the raw data loading process that requires refinement through the query designer. M notation represents the wrangling of the tabular data. For example, when performing data analysis in open source R, multiple data frames and various refinement scripts are needed. These tasks consist of a single refinement process when mapped to individual refinement components.

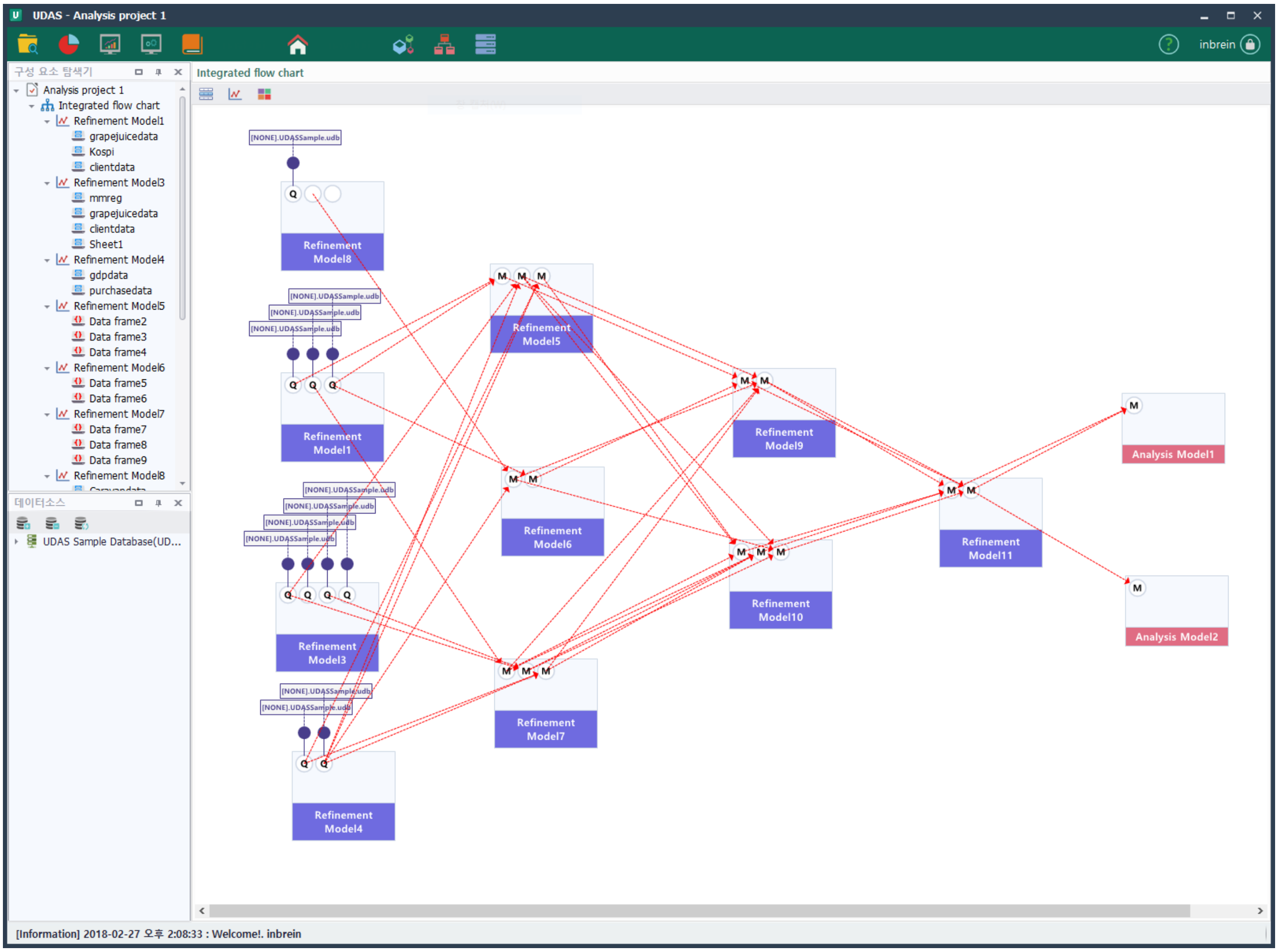
2) ANALYSIS MODEL

The analysis model component binds both statistical analysis functions and visualization libraries applicable to the data to be analyzed. Fig. 3 includes two analytical models, indicating that the refined data can be used as two different analytical models.

The refinement and analysis models included in the refinement process are executed by using the following components. Fig. 4 shows the design of the schema through ER designer, and the flow of execution of query designer to map the data to the designed schema.

i) ER Designer

ER designer designs table schemas during data refinement and performs data quality control, as shown in Fig. 5. ER designer supports various relational databases (e.g., MySQL,



[Information] 2018-02-27 오후 2:08:33 : Welcome!. inbrein

FIGURE 3. IDE screen of data refinement in the UDAS.

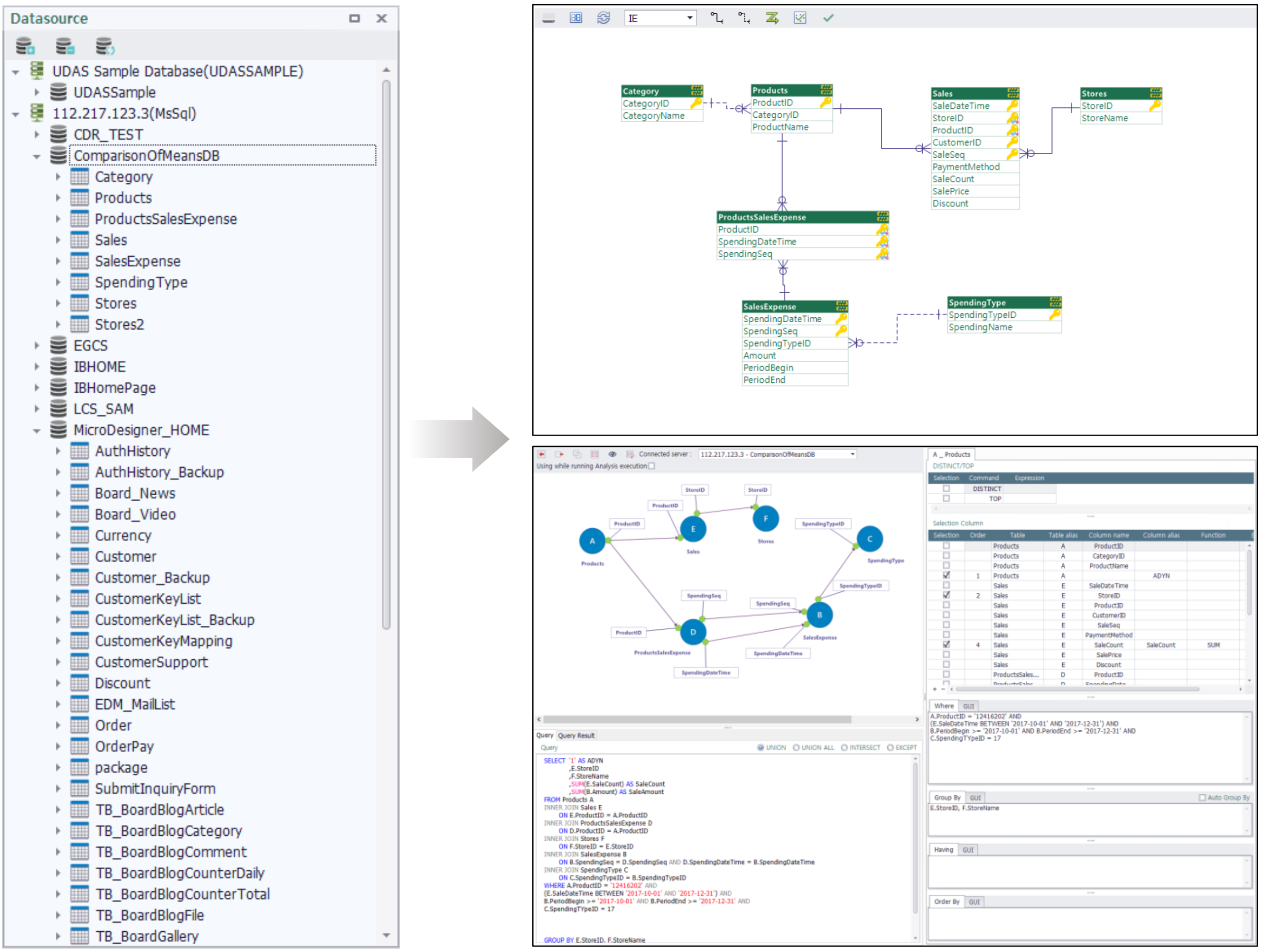
Oracle, MS-SQL, etc.) and provides forward engineering and reverse engineering for analyzed data generation to compare and merge data schemas.

ii) Query Designer

The query designer supports the design of the relationship and the schema structure of the data with the drag and drop function based on the VPL concept, and generates the SQL statement for data collection, as shown in Fig. 6. There are four IDE components as follows: Join operation & data source, Query design, Query & query result, and Input detailed information. Join operation & data source performs the join query (e.g., inner join, outer join, and cross join) between data to be refined. Query design represents the components of the FROM clause and JOIN clause that make up a query. A large circle represents a table in a FROM or JOIN clause, and a small circle represents a column used in a JOIN operation. The arrows indicate the equal operator between the table and the columns. Query & query result is the execution syntax for the query designed by the GUI and is automatically generated; the designed query can be confirmed. Input detailed information is a function with which to input various requirements such as column selection, place clause, and group by clause, where the clause constitutes the query statement in detail.

3) MAPPING DESIGNER

When mapping rules for the schema and data are defined through the mapping designer interface, the refined script is invoked to automatically perform schema-to-data mapping. At this time, data quality is guaranteed through the metadata regarding the data and schema. In addition, in-memory joins can be performed to map large amounts of data, and various types of data can be integrated by data joining between heterogeneous DBMSs. Fig. 7 shows the mapping designer interface, where nodes represent physical tables and small circles attached to nodes represent the columns of each table. The users of UDAS can specify and verify the mapping between the tables. As shown in Fig. 7, Mapping designer displays the FROM and JOIN clauses that make up the join operation through the GUI. Selected column allows the analyst to select the column to be used for the mapping.



```
SELECT '1' AS ADYN
        ,E.StoreID
        ,F.StoreName
        ,SUM(E.SaleCount) AS SaleCount
        ,SUM(B.Amount) AS SaleAmount
FROM Products A
INNER JOIN Sales E
     ON E.ProductID = A.ProductID
INNER JOIN ProductsSalesExpense D
     ON D.ProductID = A.ProductID
INNER JOIN Stores F
     ON F.StoreID = E.StoreID
INNER JOIN SalesExpense B
     ON B.SpendingSeq = D.SpendingSeq AND D.SpendingDateTime = B.SpendingDateTime
INNER JOIN SpendingType C
     ON C.SpendingTypeID = B.SpendingTypeID
WHERE A.ProductID = '12416202' AND
(E.SaleDateTime BETWEEN '2017-10-01' AND '2017-12-31') AND
B.PeriodBegin >= '2017-10-01' AND B.PeriodEnd >= '2017-12-31' AND
C.SpendingTYpeID = 17



GROUP BY E.StoreID, F.StoreName
```

Selection	Command	Expression
□	DISTINCT	
□	TOP	

Selection	Order	Table	Table alias	Column name	Column alias	Function
□		Products	A	ProductID		
□		Products	A	CategoryID		
□		Products	A	ProductName		
☑	1	Products	A		ADYN	
□		Sales	E	SaleDateTime		
☑	2	Sales	E	StoreID		
□		Sales	E	ProductID		
□		Sales	E	CustomerID		
□		Sales	E	SaleSeq		
□		Sales	E	PaymentMethod		
☑	4	Sales	E	SaleCount	SaleCount	SUM
□		Sales	E	SalePrice		
□		Sales	E	Discount		
□		ProductsSales...	D	ProductID		

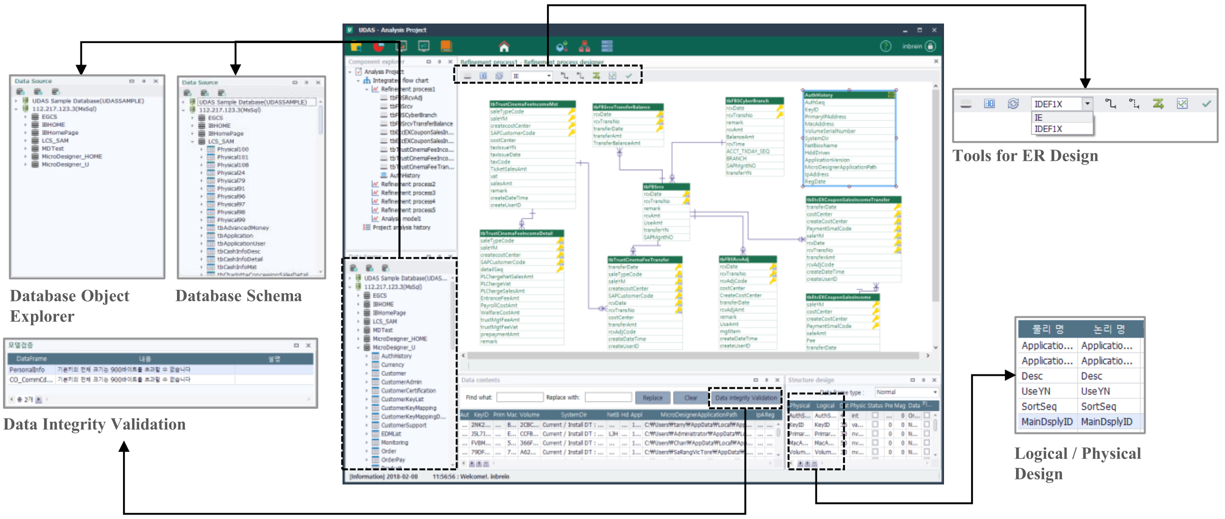
Where:

```
A.ProductID = '12416202' AND
(E.SaleDateTime BETWEEN '2017-10-01' AND '2017-12-31') AND
B.PeriodBegin >= '2017-10-01' AND B.PeriodEnd >= '2017-12-31' AND
C.SpendingTYpeID = 17
```

Group By:

```
E.StoreID, F.StoreName
```

FIGURE 4. Query design flow of data refinement in the UDAS.

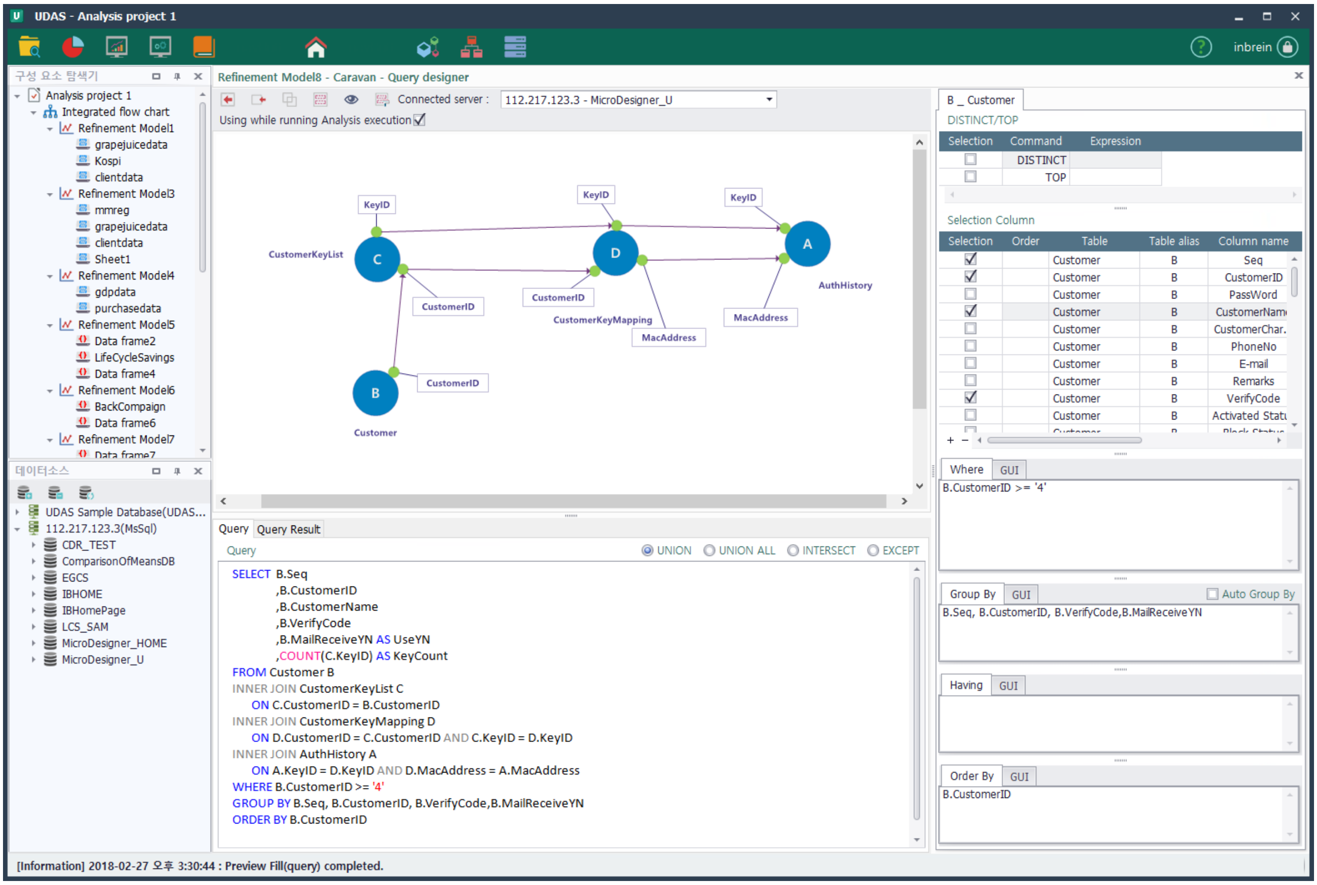


물리 명	논리 명
Applicatio...	Applicatio...
Applicatio...	Applicatio...
Desc	Desc
UseYN	UseYN
SortSeq	SortSeq
MainDsplyID	MainDsplyID

FIGURE 5. ER designer in the UDAS.

C. DATA ANALYSIS

The data analysis step derives the visualization result by running the analysis model using the refined data. Through the UDAS interface, users can understand the structure and type of refined data, produce new statistical analysis scripts, and assemble the used scripts to generate analysis models. The UDAS recommends statistical analysis libraries and functions related to the metadata of the refined data to enhance



Connected server : 112.217.123.3 - MicroDesigner_U

Using while running Analysis execution ☑

Query ● UNION ○ UNION ALL ○ INTERSECT ○ EXCEPT

```
SELECT  B.Seq
        ,B.CustomerID
        ,B.CustomerName
        ,B.VerifyCode
        ,B.MailReceiveYN AS UseYN
        ,COUNT(C.KeyID) AS KeyCount
FROM Customer B
INNER JOIN CustomerKeyList C
    ON C.CustomerID = B.CustomerID
INNER JOIN CustomerKeyMapping D
    ON D.CustomerID = C.CustomerID AND C.KeyID = D.KeyID
INNER JOIN AuthHistory A
    ON A.KeyID = D.KeyID AND D.MacAddress = A.MacAddress
WHERE B.CustomerID >= '4'
GROUP BY B.Seq, B.CustomerID, B.VerifyCode,B.MailReceiveYN
ORDER BY B.CustomerID
```

DISTINCT/TOP

Selection	Command	Expression
□	DISTINCT	
□	TOP	

Selection Column

Selection	Order	Table	Table alias	Column name
☑		Customer	B	Seq
☑		Customer	B	CustomerID
□		Customer	B	PassWord
☑		Customer	B	CustomerNam
□		Customer	B	CustomerChar.
□		Customer	B	PhoneNo
□		Customer	B	E-mail
□		Customer	B	Remarks
☑		Customer	B	VerifyCode
□		Customer	B	Activated Statu

Where: B.CustomerID >= '4'

Group By: B.Seq, B.CustomerID, B.VerifyCode,B.MailReceiveYN

Having:

Order By: B.CustomerID

[Information] 2018-02-27 오후 3:30:44 : Preview Fill(query) completed.

FIGURE 6. Query designer in the UDAS.

the user's convenience. In addition, it supports the derivation of various insights on the analysis result by recommending the visualization library applicable to the data with the data frame type derived through analysis model execution. The main components used in the analysis phase are as follows.

1) ANALYSIS MODEL DESIGNER

The UDAS implements the analysis model designer for analysis model design. The UDAS applies R, which is widely used in the statistical analysis field, and generates R script as a result of the analysis model design. When performing analysis using the R tool, considerable effort and time are required to modify and supply the script to change the raw data, library, and parameter values included in the analysis model. To solve this problem, the UDAS can assemble analysis scripts in the form of drag and drop, visualize the analysis process, and generate the entire analysis process and detailed step-by-step results. As a result, reusability and convenience of users can be improved.

2) VISUALIZATION COMPONENT

Fig. 8 shows a list of R package functions and D3.js graphs that can be selected from the analysis model. As a result of the analysis model, it is possible to confirm whether the data frame obtained can be visualized in any form (e.g., pie chart, gantt chart, time series chart, etc.). In addition, it provides single and multiple visualization functions and can perform analysis considering various aspects.

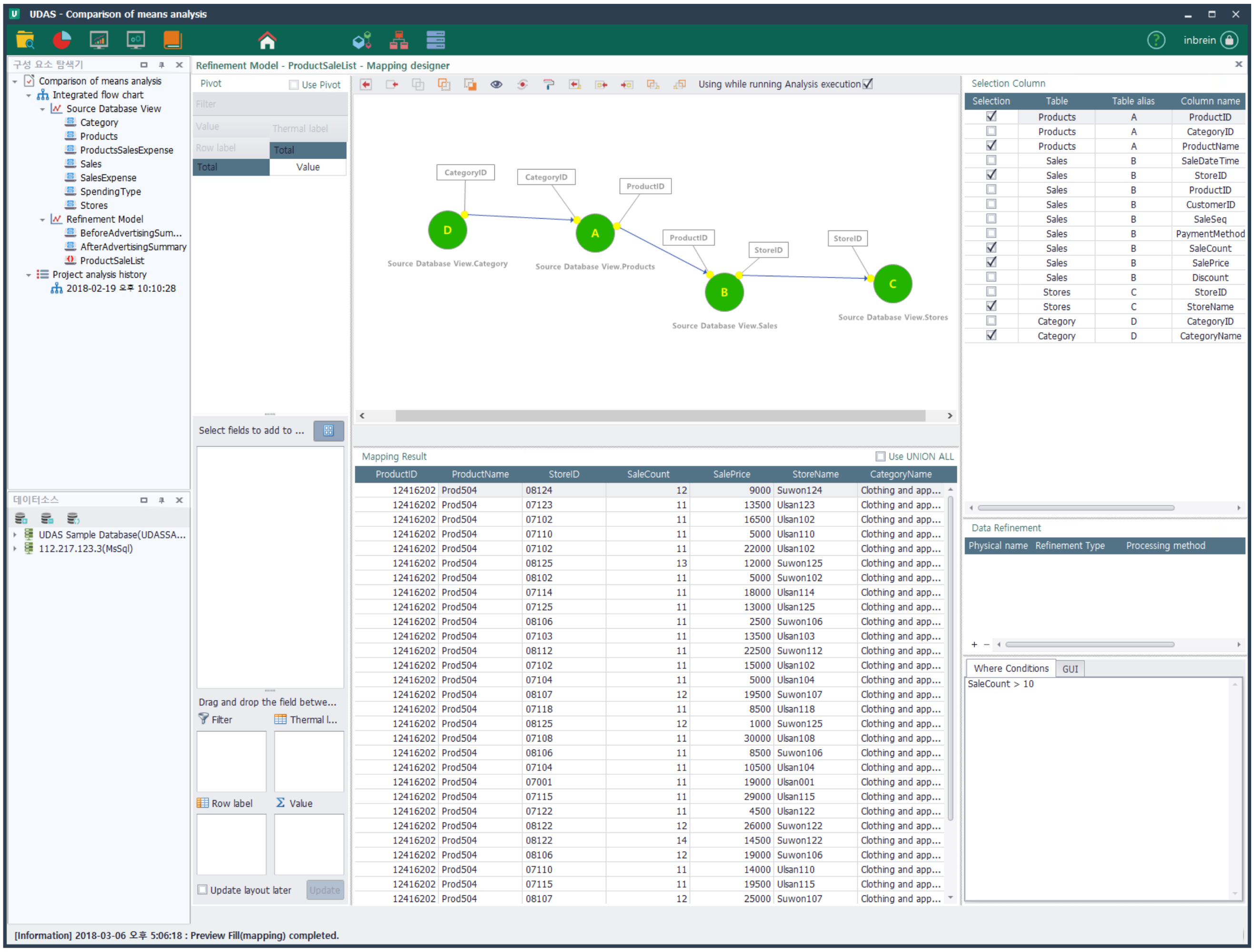
V. USE-CASE

A. APPLICATION SCENARIO AND ANALYSIS RESULTS

The UDAS is a platform-independent system and performs all processes of data analysis in a cloud environment. In this section, we show a use-case study for UDAS based on experimental data; the applied cloud computing is as follows.

1) APPLICATION SCENARIO

The scenario for performing analysis on experimental data by applying the UDAS consists of four parts as shown in Fig 9. First, a UDAS cloud system collects structured and unstructured big data. Second, the requirements are analyzed according to the purpose of the analysis model, and the refinement and analysis model is designed through the UDAS. Third, the refinement process, including designing, wrangling, and mapping is performed. Finally, the analysis process is executed and the results are visualized. Experimental data were selected from the viewpoint of verifying the function of the



Mapping Result □ Use UNION ALL

ProductID	ProductName	StoreID	SaleCount	SalePrice	StoreName	CategoryName
12416202	Prod504	08124	12	9000	Suwon124	Clothing and app...
12416202	Prod504	07123	11	13500	Ulsan123	Clothing and app...
12416202	Prod504	07102	11	16500	Ulsan102	Clothing and app...
12416202	Prod504	07110	11	5000	Ulsan110	Clothing and app...
12416202	Prod504	07102	11	22000	Ulsan102	Clothing and app...
12416202	Prod504	08125	13	12000	Suwon125	Clothing and app...
12416202	Prod504	08102	11	5000	Suwon102	Clothing and app...
12416202	Prod504	07114	11	18000	Ulsan114	Clothing and app...
12416202	Prod504	07125	11	13000	Ulsan125	Clothing and app...
12416202	Prod504	08106	11	2500	Suwon106	Clothing and app...
12416202	Prod504	07103	11	13500	Ulsan103	Clothing and app...
12416202	Prod504	08112	11	22500	Suwon112	Clothing and app...
12416202	Prod504	07102	11	15000	Ulsan102	Clothing and app...
12416202	Prod504	07104	11	5000	Ulsan104	Clothing and app...
12416202	Prod504	08107	12	19500	Suwon107	Clothing and app...
12416202	Prod504	07118	11	8500	Ulsan118	Clothing and app...
12416202	Prod504	08125	12	1000	Suwon125	Clothing and app...
12416202	Prod504	07108	11	30000	Ulsan108	Clothing and app...
12416202	Prod504	08106	11	8500	Suwon106	Clothing and app...
12416202	Prod504	07104	11	10500	Ulsan104	Clothing and app...
12416202	Prod504	07001	11	19000	Ulsan001	Clothing and app...
12416202	Prod504	07115	11	29000	Ulsan115	Clothing and app...
12416202	Prod504	07122	11	4500	Ulsan122	Clothing and app...
12416202	Prod504	08122	12	26000	Suwon122	Clothing and app...
12416202	Prod504	08122	14	14500	Suwon122	Clothing and app...
12416202	Prod504	08106	12	19000	Suwon106	Clothing and app...
12416202	Prod504	07110	11	14000	Ulsan110	Clothing and app...
12416202	Prod504	07115	11	19500	Ulsan115	Clothing and app...
12416202	Prod504	08107	12	25000	Suwon107	Clothing and app...

Selection	Table	Table alias	Column name
☑	Products	A	ProductID
□	Products	A	CategoryID
☑	Products	A	ProductName
□	Sales	B	SaleDateTime
☑	Sales	B	StoreID
□	Sales	B	ProductID
□	Sales	B	CustomerID
□	Sales	B	SaleSeq
□	Sales	B	PaymentMethod
☑	Sales	B	SaleCount
☑	Sales	B	SalePrice
□	Sales	B	Discount
□	Stores	C	StoreID
☑	Stores	C	StoreName
□	Category	D	CategoryID
☑	Category	D	CategoryName

SaleCount > 10

[Information] 2018-03-06 오후 5:06:18 : Preview Fill(mapping) completed.

FIGURE 7. Mapping designer in the UDAS.

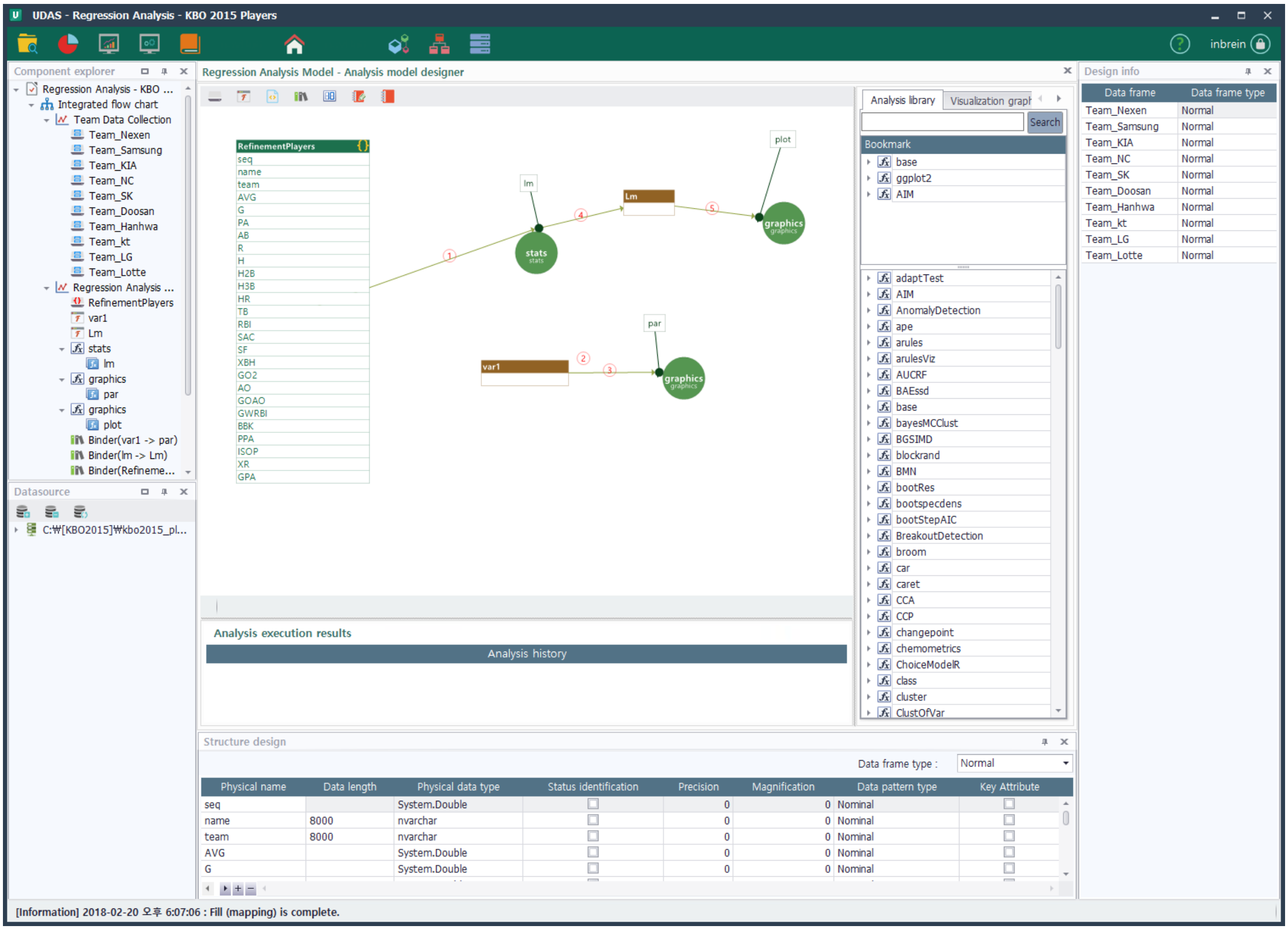
UDAS that can be analyzed in terms of unstructured text and formal data. Experimental data were collected from e-mail and log data over a total of 60 months from a company that provides e-mail services on a 50 TB scale. Through the experimental data, we demonstrate that the UDAS is capable of analyzing unstructured and structured data accurately and show that flexible and various visualization libraries can be applied in the UDAS. Fig. 10 shows the analysis process of the UDAS in the use-case study.

2) ANALYSIS RESULTS

Fig. 11 shows the results of the analysis model. The analysis of the e-mail and log data shows that four different charts can be generated. For the purpose of analysis, the UDAS diagnoses the type of data and recommends visualization libraries applicable to the data. The upper left chart of Fig. 11 shows the amount of mails received by each department for the specified period from 2011 to 2015 is calculated as a result of analyzing all mails. A small circle represents the amount of mail received at a specific time in one department. The upper right chart of Fig. 11 shows the number of unethical words used in each year. The frequency of words, which are registered in the unethical word dictionary and used in e-mail over 60 months, is integrated for each department. The five departments with the highest frequencies are selected and visualized in the form of a bar chart. The bottom left chart of Fig. 11 shows the results of analytical model that compile the number of violations of departmental security policy for each year. The top five departments with the total number of composed e-mails violating the security policy are extracted. The bottom right chart of Fig. 11 is the number and capacity of individual mailings for a specific period of time. The blue represents the external mail, and the red the in-house mail. It shows the number and capacity of the top ten people by usage.

B. QUALITATIVE EVALUATION

This clause describes the method and results of the user satisfaction evaluation for the UDAS. The UDAS is a big data analysis platform that includes the collection, refinement, processing, and analysis of structured and unstructured



Data frame	Data frame type
Team_Nexen	Normal
Team_Samsung	Normal
Team_KIA	Normal
Team_NC	Normal
Team_SK	Normal
Team_Doosan	Normal
Team_Hanhwa	Normal
Team_kt	Normal
Team_LG	Normal
Team_Lotte	Normal

Data frame type : Normal

Physical name	Data length	Physical data type	Status identification	Precision	Magnification	Data pattern type	Key Attribute
seq		System.Double	□	0	0	Nominal	□
name	8000	nvarchar	□	0	0	Nominal	□
team	8000	nvarchar	□	0	0	Nominal	□
AVG		System.Double	□	0	0	Nominal	□
G		System.Double	□	0	0	Nominal	□

[Information] 2018-02-20 오후 6:07:06 : Fill (mapping) is complete.

FIGURE 8. Analysis model designer in the UDAS.

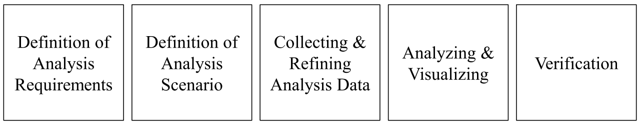
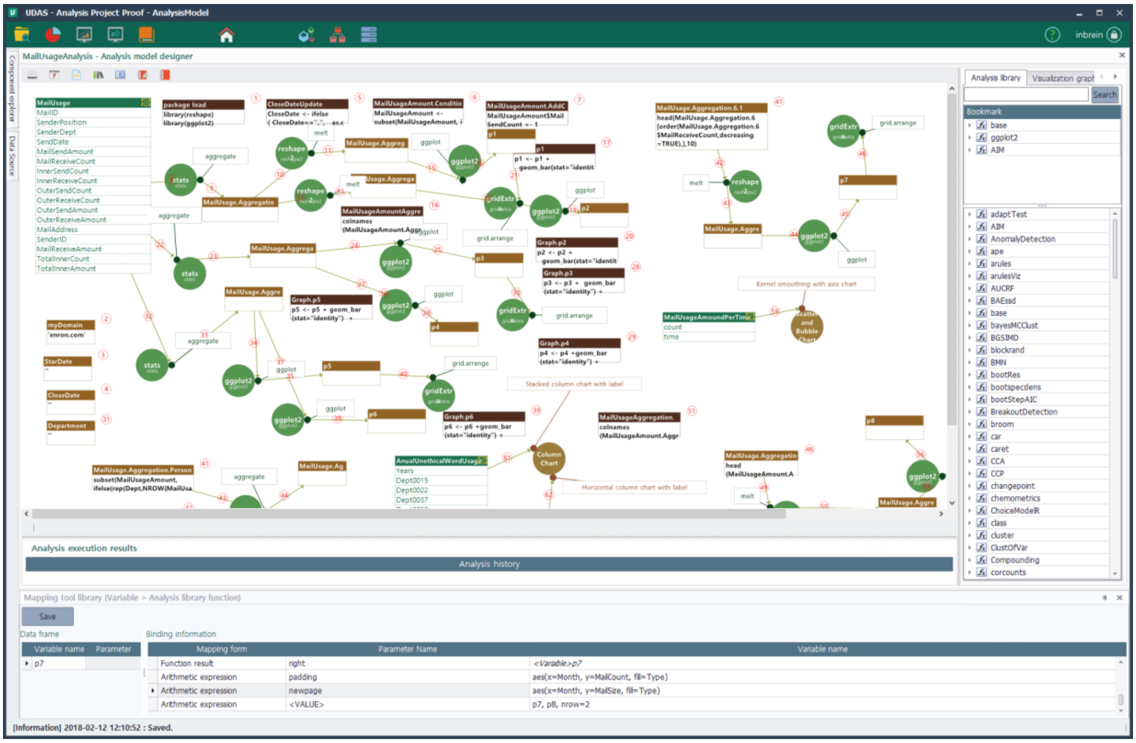

Definition of Analysis Requirements
Definition of Analysis Scenario
Collecting & Refining Analysis Data
Analyzing & Visualizing
Verification

FIGURE 9. Analysis steps.

text data. Therefore, evaluation factors for the representative functions of the big data analysis platform are selected, and a user satisfaction survey is conducted on the UDAS based on the evaluation factors. ISO/IEC 25010 is an international standard that defines the quality evaluation factors for product quality evaluation [44]–[46]. Based on the evaluation factors defined in ISO/IEC 25010, the quality evaluation of data and the quality evaluation of the big data platform are performed [47], [48]. Therefore, we evaluate user satisfaction by applying the quality evaluation factor defined in ISO/IEC 25010 for usability evaluation of the functional elements of the UDAS. Fig. 12 shows the evaluation factors defined by ISO/IEC 25010.

The questionnaire items are designed based on the evaluation factors, and a usability evaluation of the UDAS was performed on three sample groups, namely data analysts, developers, and ordinary users. The Cronbach alpha coefficient is widely used to measure the reliability of questionnaire items [49]–[52]. The reliability of the questionnaire was confirmed by the Cronbach alpha coefficient of 0.82. A total of 60 respondents participated in the usability test for the UDAS, and we calculated the average value of the evaluation factors and converted it to 100. As a result, the user satisfaction score for the UDAS system as a big data analysis system was 81.2.

Table 2 shows the usability evaluation results for each evaluation factor. As shown in Table 2, the UDAS has a relatively high score in terms of usability, reliability, maintainability, portability, functional suitability, compatibility, efficiency, and security. Through the UDAS interface with the VPL concept, the usability score confirms that the user can easily design the refinement and analysis model. In addition, the data refinement process and analysis model can be saved as a template. It means that users reuse an appropriate template for the next time, in other words, it indicates that the reproducibility of past analysis projects is possible. Further, through functions such as ER-designer and mapper in the



Mapping tool library (Variable > Analysis library function)

Save

Data frame

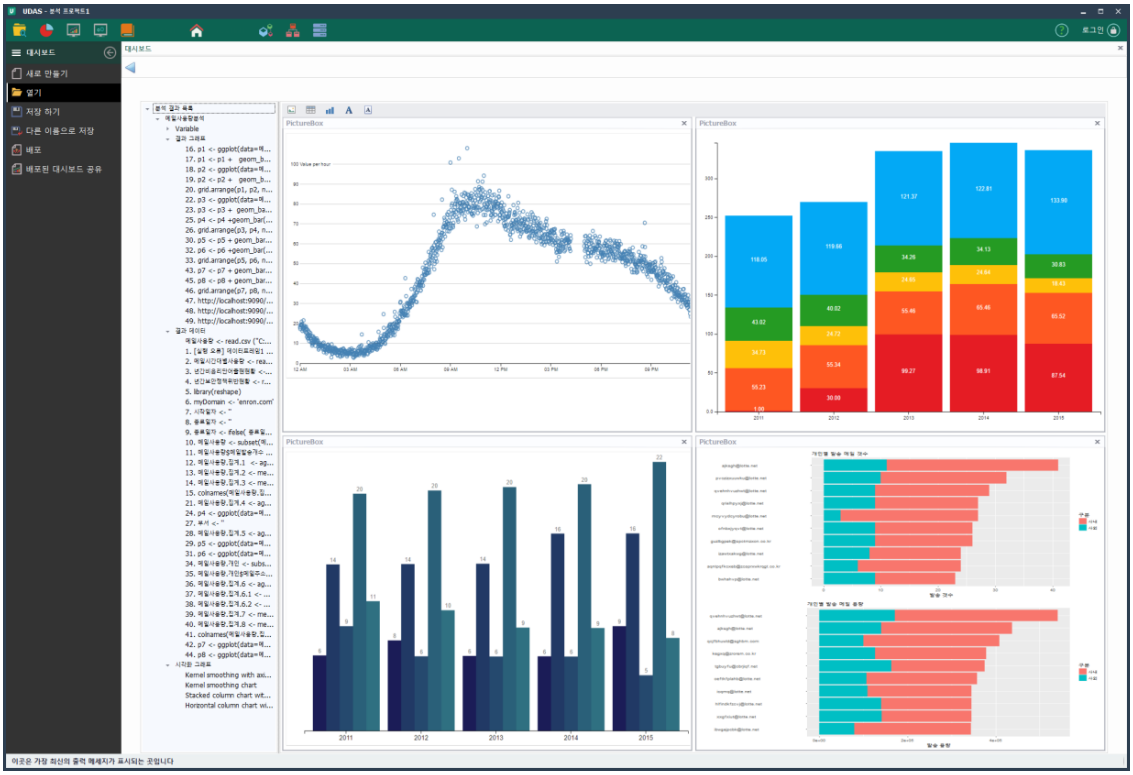
Variable name	Parameter
p7	

Binding information

Mapping form	Parameter Name	Variable name
Function result	right	<Variable>p7
Arithmetic expression	padding	aes(x=Month, y=MailCount, fill=Type)
Arithmetic expression	newpage	aes(x=Month, y=MailSize, fill=Type)
Arithmetic expression	<VALUE>	p7, p8, nrow=2

[Information] 2018-02-12 12:10:52 : Saved.

FIGURE 10. Analysis process for use case in the UDAS.



- 분석 결과 목록
  - 메일사용량분석
    - Variable
    - 결과 그래프
      - 16. p1 <- ggplot(data=메...
      - 17. p1 <- p1 + geom_b...
      - 18. p2 <- ggplot(data=메...
      - 19. p2 <- p2 + geom_b...
      - 20. grid.arrange(p1, p2, n...
      - 22. p3 <- ggplot(data=메...
      - 23. p3 <- p3 + geom_ba...
      - 25. p4 <- p4 +geom_bar(...
      - 26. grid.arrange(p3, p4, n...
      - 30. p5 <- p5 + geom_bar...
      - 32. p6 <- p6 +geom_bar(...
      - 33. grid.arrange(p5, p6, n...
      - 43. p7 <- p7 + geom_bar...
      - 45. p8 <- p8 + geom_bar...
      - 46. grid.arrange(p7, p8, n...
      - 47. http://localhost:9090/...
      - 48. http://localhost:9090/...
      - 49. http://localhost:9090/...
    - 결과 데이터
      - 메일사용량 <- read.csv ("C:...
      - 1. [실행 오류] 데이터프레임1 ...
      - 2. 메일시간대별사용량 <- rea...
      - 3. 년간비윤리단어출현현황 <-...
      - 4. 년간보안정책위반현황 <- r...
      - 5. library(reshape)
      - 6. myDomain <- 'enron.com'
      - 7. 시작일자 <- ''
      - 8. 종료일자 <- ''
      - 9. 종료일자 <- ifelse(종료일...
      - 10. 메일사용량 <- subset(메...
      - 11. 메일사용량$메일발송개수 ...
      - 12. 메일사용량.집계.1 <- ag...
      - 13. 메일사용량.집계.2 <- me...
      - 14. 메일사용량.집계.3 <- me...
      - 15. colnames(메일사용량.집...
      - 21. 메일사용량.집계.4 <- ag...
      - 24. p4 <- ggplot(data=메...
      - 27. 부서 <- ''
      - 28. 메일사용량.집계.5 <- ag...
      - 29. p5 <- ggplot(data=메...
      - 31. p6 <- ggplot(data=메...
      - 34. 메일사용량.개인 <- subs...
      - 35. 메일사용량.개인$메일주소...
      - 36. 메일사용량.집계.6 <- ag...
      - 37. 메일사용량.집계.6.1 <- ...
      - 38. 메일사용량.집계.6.2 <- ...
      - 39. 메일사용량.집계.7 <- me...
      - 40. 메일사용량.집계.8 <- me...
      - 41. colnames(메일사용량.집...
      - 42. p7 <- ggplot(data=메...
      - 44. p8 <- ggplot(data=메...
    - 시각화 그래프
      - Kernel smoothing with axi...
      - Kernel smoothing chart
      - Stacked column chart wit...
      - Horizontal column chart wi...

이곳은 가장 최신의 출력 메세지가 표시되는 곳입니다

FIGURE 11. Results of use case evaluation.

data refinement step, it is possible to prevent the loss of raw data and to enable the integration of schema design, thereby assuring the reliability of the statistical analysis result of the UDAS. It is confirmed that various wrangling operations are performed in collecting and refining large amounts of unstructured text data and structured data, and the efficiency

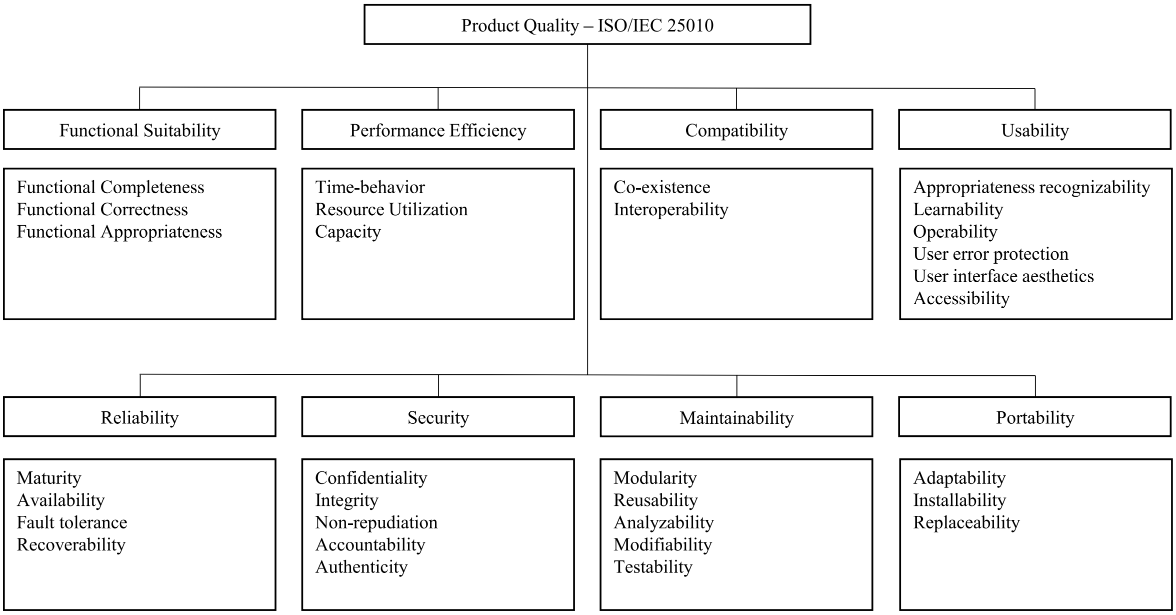


FIGURE 12. Evaluation factors of product quality.

TABLE 2. Evaluation factors and results.

Evaluation Factor	Description	Evaluation Result
Functional Suitability	Degree to which a system provides functions that meet stated and implied needs when used under specified conditions	3.99
Performance Efficiency	Performance relative to the amount of resources used under stated conditions	3.73
Compatibility	Degree to which a system or component can exchange information with other products, systems, or components	3.82
Usability	Degree to which a product or system can be used by specified users to achieve specified goals with effectiveness, efficiency, and satisfaction in a specified context of use	4.51
Reliability	Degree to which a system, product, or component performs specified functions under specified conditions for a specified period of time	4.48
Security	Degree to which a system protects information and data so that persons or other systems have the degree of data access appropriate to their type and level of authorization	3.47
Maintainability	Degree of effectiveness and efficiency with which a system can be modified by the intended maintainers	4.38
Portability	Degree of effectiveness and efficiency with which a system or component can be transferred from one software or usage environment to another	4.13

of performance is relatively low in integrating various data into the analysis model. However, this work is a necessary step in the design process of the refinement and analytical model and requires expert knowledge. Therefore, the UDAS focuses more on the analytical reliability and functional usability of the system. The UDAS was also developed as a web-based interface and executed as a platform independent system in a cloud computing based on the big data ecosystem, so that it can collect and process large amounts of data reliably. In addition, it is easy to design and execute the analysis process and support various analyses of the analysis results data by applying various visualization libraries.

VI. CONCLUSION

Data refinement plays an important role in the process of big data analysis to ensure the accuracy of the analysis results and derive insights from big data. However, existing systems and tools require either unnecessary iterative tasks or considerable time and effort for the process of data refinement. We propose the UDAS system. The UDAS system creates a template for each purification purpose by linking data collection and preprocessing operations that performed in the data refinement process. By using the generated template, it is possible to perform efficiently the refinement process that requires the most time in big data analysis. The UDAS system also provides an interface with the VPL concept to design data collection, preprocessing, and integration models necessary for data analysis as well as data analysis. For data analysis, UDAS can perform customized analysis not only for libraries provided by R in conjunction with open source R but also for user's purpose. The UDAS system also provides a preview of the visualization results that can be derived from the data type and analysis library, allowing the user to confirm the analysis method design. In other words, the UDAS system has various functions such as data collection, refinement, and analysis necessary for performing big data analysis. UDAS system has an advantage that it is executed independently in the operating environment through the cloud computing. As a result, we obtained 81.2 usability evaluations for the UDAS system for data analysis experts, domain experts, and general users.

REFERENCES

[1] I. A. T. Hashem *et al.*, "The rise of 'big data' on cloud computing: Review and open research issues," *Inf. Syst.*, vol. 47, pp. 98–115, Jan. 2015.

[2] J. Scott, *Social Network Analysis: A Handbook*, 2nd ed. London, U.K.: Sage, 2017.

[3] X. Jin, B. W. Wah, X. Cheng, and Y. Wang, "Significance and challenges of big data research," *Big Data Res.*, vol. 2, no. 2, pp. 59–64, 2015.

[4] Z. Khan, A. Anjum, and S. L. Kiani, "Cloud based big data analytics for smart future cities," in *Proc. IEEE/ACM UCC*, Dresden, Germany, Dec. 2013, pp. 381–386.

[5] J. S. Ward and A. Barker. (2013). "Undefined by data: A survey of big data definitions." [Online]. Available: https://arxiv.org/abs/1309.5821

[6] A. Gandomi and M. Haider, "Beyond the hype: Big data concepts, methods, and analytics," *Int. J. Inf. Manage.*, vol. 35, no. 2, pp. 137–144, 2015.

[7] A. Katal, M. Wazid, and R. H. Goudar, "Big data: Issues, challenges, tools and good practices," in *Proc. IEEE IC3*, Noida, India, Aug. 2013, pp. 404–409.

[8] R. Mikut and M. Reischl, "Data mining tools," in *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 1. Hoboken, NJ, USA: Wiley, 2011, no. 5, pp. 431–443.

[9] *Data Refinement: The Dirty Side of Data Science*. Accessed: Jul. 22, 2018. [Online]. Available: https://insidebigdata.com/2015/06/15/data-refinement-the-dirty-side-of-data-science/

[10] W. Olsen, *Data Collection: Key Debates and Methods in Social Research*. London, U.K.: Sage, 2011.

[11] R. Ranjan, "Streaming big data processing in datacenter clouds," *IEEE Cloud Comput.*, vol. 1, no. 1, pp. 78–83, May 2014.

[12] N. Garg, *Apache Kafka*. Birmingham, U.K.: Packt, 2013.

[13] B. Daniel, "Big data and analytics in higher education: Opportunities and challenges," *Brit. J. Educ. Technol.*, vol. 46, no. 5, pp. 904–920, 2015.

[14] S. LaValle, E. Lesser, R. Shockley, M. S. Hopkins, and N. Kruschwltz, "Big data, analytics and the path from insights to value," *MIT Sloan Manage. Rev.*, vol. 52, no. 2, pp. 21–32, 2011.

[15] D. Boyd and K. Crawford, "Critical questions for big data: Provocations for a cultural, technological, and scholarly phenomenon," *Inf., Commun. Soc.*, vol. 15, no. 5, pp. 662–679, 2012.

[16] P. Russom, "Big data analytics," *TDWI Best Pract. Rep., Fourth Quart.*, vol. 19, no. 4, pp. 1–34, 2011.

[17] T. Furche, G. Gottlob, L. Libkin, G. Orsi, and N. W. Paton, "Data wrangling for big data: Challenges and opportunities," in *Proc. EDBT*, Bordeaux, France, 2016, pp. 473–478.

[18] H. Chen, R. H. L. Chiang, and V. C. Storey, "Business intelligence and analytics: From big data to big impact," *MIS Quart.*, vol. 36, no. 4, pp. 1165–1188, Dec. 2012.

[19] S. F. Wamba, S. Akter, A. Edwards, G. Chopin, and D. Gnanzou, "How 'big data' can make big impact: Findings from a systematic review and a longitudinal case study," *Int. J. Prod. Econ.*, vol. 165, pp. 234–246, Jul. 2015.

[20] P. Zikopoulos and C. Eaton, *Understanding Big Data: Analytics for Enterprise Class Hadoop and Streaming Data*. New York, NY, USA: McGraw-Hill, 2011.

[21] R. Verborgh and M. De Wilde, *Using OpenRefine*. Birmingham, U.K.: Packt, 2013.

[22] D. D. Hils, "Visual languages and computing survey: Data flow visual programming languages," *J. Vis. Lang., Comput.*, vol. 3, no. 1, pp. 69–101, 1992.

[23] M. Bostock, *D3. Js, Data Driven Documents*. Birmingham, U.K.: Packt Publishing Ltd., 2012.

[24] R Development Core Team, "R: A language and environment for statistical computing," in *R Foundation for Statistical Computing*. Vienna, Austria, 2013. [Online]. Available: http://www.R-project.org

[25] *SAS/STAT 9.1 User's Guide the Reg Procedure: (Book Excerpt)*, SAS Institute Inc., Cary, NC, USA, 2008.

[26] J. Pallant and S. S. Manual, *A Step by Step Guide to Data Analysis Using SPSS*. Berkshire, U.K.: McGraw-Hill, 2010.

[27] I. Fellows, "Deducer: A data analysis GUI for R," *J. Statist. Softw.*, vol. 49, no. 8, pp. 1–15, 2012.

[28] *RStudio: Integrated Development for R*, RStudio Inc., Boston, MA, USA, 2015. [Online]. Available: http://www.rstudio.com

[29] S. Rödiger, T. Friedrichsmeier, P. Kapat, and M. Michalke, "RKWard: A comprehensive graphical user interface and integrated development environment for statistical analysis with R," *J. Statist. Softw.*, vol. 49, no. 9, pp. 1–34, 2012.

[30] M. Helbig, M. Theus, and S. Urbanek, "JGR—A Java GUI for R," *Comput. Graph. Newslett.*, vol. 16, no. 2, pp. 9–12, 2005.

[31] J. Fox, "The R commander: A basic-statistics graphical user interface to R," *J. Statist. Softw.*, vol. 14, no. 9, pp. 1–42, 2005.

[32] G. J. Williams, "Rattle: A data mining GUI for R," *R J.*, vol. 1, no. 2, pp. 45–55, 2009.

[33] S. Venkataraman *et al.*, "SparkR: Scaling R programs with spark," in *Proc. Int. Conf. Manage. Data*, 2016, pp. 1099–1104.

[34] C. Boettiger and D. Eddelbuettel, "An introduction to rocker: Docker containers for R," *J. R*, vol. 9, no. 2, pp. 527–536, 2017.

[35] N. B. Robbins and J. Effective, "Effective graphs with Microsoft R Open," Microsoft Corp., Redmond, WA, USA, Data Treatment White Paper, 2016. [Online]. Available: http://www.kudwi.com/wp-content/uploads/2016/05/effectivegraphsmro1.pdf

[36] D. Allen, "Practical aspects of R in finance," *J. Manage. Inf. Decis. Sci.*, vol. 20, pp. 1–10, Dec. 2017.

[37] G. Han, Y.-H. Kim, D. Shin, Y. Lee, and J. Seo, "Design and development of visual analytic tools in SRC-STAT," in *Proc. HCI Korea*, 2014, pp. 229–231.

[38] M. Barnett *et al.*, "Stat!: An interactive analytics environment for big data," in *Proc. ACM SIGMOD*, New York, NY, USA, 2013, pp. 1013–1016.

[39] F. B. Viegas, M. Wattenberg, F. V. Ham, J. Kriss, and M. McKeon, "ManyEyes: A site for visualization at Internet scale," *IEEE Trans. Vis. Comput. Graphics*, vol. 13, no. 6, pp. 1121–1128, Nov. 2007.

[40] M. Copeland, J. Soh, A. Puca, M. Manning, and D. Gollob, "Microsoft azure and cloud computing," in *Microsoft Azure*. Berkeley, CA, USA: Apress, 2015, pp. 3–26.

[41] J. Demšar, B. Zupan, G. Leban, and T. Curk, "Orange: From experimental machine learning to interactive data mining," in *Proc. Eur. Conf. Princ. Data Mining Knowl. Discovery*, Berlin, Germany, 2004, pp. 537–539.

[42] G. Bakos, *KNIME Essentials*. Birmingham, U.K.: Packt, 2013.

[43] *IBM Cognos Analytics*. Accessed: Jul. 22, 2018. [Online]. Available: https://www.ibm.com/software/in/analytics/cognos/platform/

[44] N. Bevan, "Extending quality in use to provide a framework for usability measurement," in *Proc. Hum. Centered Design*. Berlin, Germany: Springer, 2009, pp. 3247–3251.

[45] N. Bevan, "Usability," in *Encyclopedia of Database Systems*, L. Liu and M. T. Ž̈su, Ed. Boston, MA, USA: Springer, 2009, pp. 3247–3251.

[46] A. Hussain and E. O. C. Mkpojiogu, "An application of ISO/IEC 25010 standard in the quality-in-use assessment of an online health awareness system," *J. Teknologi*, vol. 77, no. 5, pp. 9–13, 2015.

[47] M. Jorge, C. Ismael, R. Bibiano, S. Manuel, and P. Mario, "A data quality in use model for big data," *Future Gener. Comput. Syst.*, vol. 63, pp. 123–130, Oct. 2016.

[48] S. Ouhbi, A. Idri, J. L. Fernández-Alemán, A. Toval, and H. Benjelloun, "Applying ISO/IEC 25010 on mobile personal health records," in *Proc. BIOSTEC*, Lisbon, Portugal, 2015, pp. 381–386.

[49] L. J. Cronbach, "Coefficient alpha and the internal structure of tests," *Psychometrika*, vol. 16, no. 3, pp. 297–334, 1951.

[50] J. R. A. Santos, "Cronbach's alpha: A tool for assessing the reliability of scales," *J. Extension*, vol. 37, no. 2, pp. 1–5, 1999.

[51] O. Kwon, N. Lee, and B. Shin, "Data quality management, data usage experience and acquisition intention of big data analytics," *Int. J. Inf. Manage.*, vol. 34, no. 3, pp. 387–394, 2014.

[52] R. Schmidt, M. Möhring, R.-C. Härting, C. Reichstein, P. Neumaier, and P. Jozinović, *Industry 4.0—Potentials for Creating Smart Products: Empirical Research Results*, W. Abramowicz, Ed. Cham, Switzerland: Springer, 2015, pp. 16–27.

HYUNJIN CHOI received the B.S. degree in computer engineering from Hannam University, Daejeon, South Korea, in 1991, and the M.S. degree in software engineering from Korea University, Seoul, South Korea, in 2006, where he is currently pursuing the Ph.D. degree with the Software System Laboratory, Department of Computer and Radio Communications Engineering. From 2007 to 2010, he was an Adjunct Professor in computer science with the Cyber University of Korea, Seoul. From 2012 to 2013, he was an Adjunct Professor in computer software engineering with Sangmyung University. His research interests include software engineering, data analysis, data modeling and refinement, data visualization, and robotics process automation.

JANGWON GIM received the B.S. degree in computer software engineering from Sangmyung University, South Korea, in 2005, and the M.S. and Ph.D. degrees in computer and radio communications engineering from Korea University in 2009 and 2013, respectively. From 2013 to 2017, he was a Senior Researcher with the Korea Institute of Science and Technology. He is currently an Assistant Professor with the Department of Software Convergence Engineering, Kunsan National University. His research interests include text mining, urban environment analysis, patent analysis, and semantic Web for artificial intelligence. He has been a Committee Member of ISO/IEC JTC1/SC32 for six years.

YOUNG-DUK SEO received the B.S. degree in computer and communication engineering and the Ph.D. degree in computer science and engineering from Korea University, Seoul, South Korea, in 2012 and 2018, respectively. From 2012 to 2017, he was a member of the Center of Autonomous and Adaptive Software, Korea University, where he is currently a Research Professor with the Computer, Information and Communication Research Institute. His research interests include self-adaptive software, big data analysis, recommender systems, and entity linking.

DOO-KWON BAIK received the B.S. degree in mathematics from Korea University, Seoul, South Korea, in 1974, and the M.S. and Ph.D. degrees in computer science from Wayne State University, Detroit, MI, USA, in 1983 and 1986, respectively. He was the Founder and Director of the Information and Communication Research Institute, Korea University, where he is currently the Director of the Software System Laboratory and a Full Professor with the Department of Computer Science and Engineering. His research interests include data engineering, software engineering, metadata registry, and modeling and simulation. He has been a Committee Member of ISO/IEC JTC1/SC32 for 20 years, and he was the Chair of ISO/IEC JTC1/SC32-Korea.