Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 207 (2022) 360–369

Procedia
Computer Science

www.elsevier.com/locate/procedia

26th International Conference on Knowledge-Based and Intelligent Information & Engineering Systems (KES 2022)

Three-Step Master Data Creation Method from Big Data: Scraping, Semi-Structuring, and Extraction

Tsukasa Kudo[a,*], Takehiro Yamamoto[a], Tomoki Watanabe[a]

[a]*Shizuoka Institute of Science and Technology, 2200-2 Toyosawa, Fukuroi, Shizuoka 437-8555, Japan*

Abstract

Master data preparation for the introduction of business systems is often a burden. Nevertheless, a huge amount of various data is released as big data and used in several fields. However, these data exist in various formats at various websites on big data, making it difficult to utilize them directly for preparing master data. In this study, we propose a three-step master creation method: necessary data are acquired by scraping, a semi-structured common database is created by integrating these data, and individual master data is extracted from this database. Furthermore, we evaluate this proposed method by creating an English question master for e-learning and show that individual master data can be efficiently extracted from the integrated common database. Additionally, we conducted a comparative evaluation of the efficiency of the proposed method using MongoDB and MySQL, namely a semi-structured and structured database. The experimental results show that the semi-structured database is more efficient for manipulating data of various websites.

Keywords: Scraping, Big data, Master data, MongoDB, Semi-structured database, e-learning

1. Introduction

Currently, in organizations, such as companies, many businesses are conducted by information systems (hereinafter, business systems), and their data plays a central role. Business systems' data is divided into master data (hereinafter, master) and transaction data. For example, in a sales management system, product and customer information is classified as master; sales record is classified as transaction data. Master is defined as the basic characteristics of instances of business entities, and once created, it is not changed frequently. It has been mentioned that masters' quality is necessary to achieve the purpose of the business system [9].

* Corresponding author. Tel.: +81-538-45-0201 ; fax: +81-538-45-0110.
E-mail address: kudo.tsukasa@sist.ac.jp

10.1016/j.procs.2022.09.070

Master creation is a part of the business system introduction process, and it is necessary to complete this process to operate the system. However, the introduction process often involves a heavy load to collecting various data and setting up a master, and it becomes an obstacle to introducing the system.

In contrast, nowadays, a huge amount of various data is released as big data, and it is used in several applications, such as analysis by data mining [15]. Similarly, standardized data is released as master. For example, standardized data on book information is published as international standard book numbers (ISBNs), and book information can be obtained from ISBNs to facilitate the preparation of masters.

However, the masters used in individual business systems are not always consistent with such standardized data. In addition, even when using standardized data such as ISBNs mentioned above, it is necessary to extract necessary data from various websites for actual business system operations. For example, it is necessary to refer to the publisher's website for detailed information on books, to various review sites for evaluating books for purchasing, and to the company's sales data for grasping sales trends for product placement decisions. In other words, it is necessary to generate the necessary information by integrating data from various sites and to update the master in response to sales trends and new books even after the master is generated. Furthermore, given that each website is independently produced and operated, the data format and attributes of each website are different. So, it has been pointed out that the problem of sparse information occurs when such information is integrated using a relational database [12].

In this study, we propose a three-step master creation method to efficiently create individual business systems' masters from big data. First, we acquire individual data from each website; Second, we integrate the acquired data into a semi-structured common database for efficient retrieval of the necessary information while avoiding the above-mentioned problem; Third, we create a target individual master by extracting the necessary data from the common database.

Furthermore, we evaluate the efficiency of the proposed method by creating an English question master for e-learning, which provides appropriate questions to individual learners by utilizing information from various websites. It shows that the efficiency can be improved by sharing the first two steps, which create the common database to extract individual masters in the third step. Additionally, to show that a semi-structured database is valid as the common database, we conduct comparative efficiency evaluations between a semi-structured and structured database using MongoDB and MySQL, respectively. It shows that the former is more efficient in handling various data acquired from many websites in an integrated manner.

The remainder of this paper is organized as follows. Section 2 describes the related works and the issues encountered in creating individual masters from big data. Section 3 describes the three-step master creation method. Section 4 describes the implementation of the English question master creation system. Section 5 evaluates the implemented method. Section 6 discusses the evaluation results. Finally, Sec. 7 gives a conclusion.

2. Related works and issues to creating masters from big data

Nowadays, a huge amount of diverse data from various data sources is shared as big data and used for recommendation systems, search engines, and analysis of various information [15]. Additionally, it is underway to utilize such big data in various fields, such as infectious diseases and education [1, 14].

Since big data has the following characteristics: volume, variety, and velocity, various issues have been mentioned when handling it in relational databases. Also, various Not only SQL (NoSQL) databases, including MongoDB, and New SQL databases have been used [12, 13]. MongoDB is a document-based database system, in which data are represented by documents. Such data are semi-structured data in the form of Binary JSON (BSON). So, the collections that store documents, which correspond to tables in relational databases, can store various data. That is, such collections are suitable for storing and managing various format data on big data [13]. Also, it has been shown that such collections are more efficient than relational databases such as MySQL for most data operations [11].

In this paper, to unify the terms, we use the term "collection" for also tables in relational databases and "document" for also records.

Various methods are used to improve business efficiency by standardizing data and providing it as cloud services. For example, for books, ISBN uniquely identifies each published book, including its edition. ISBN is managed by the National ISBN Agency of each country. In Japan, book search services are provided by the book information database managed by the publishing industry[10]. Similarly, product information is provided by a product database based on

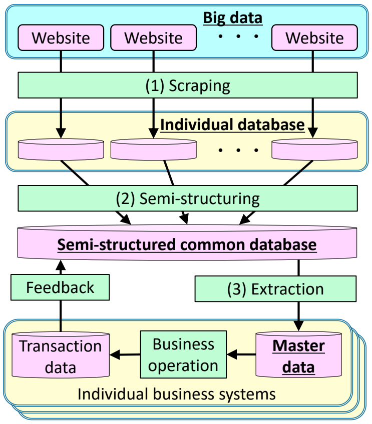
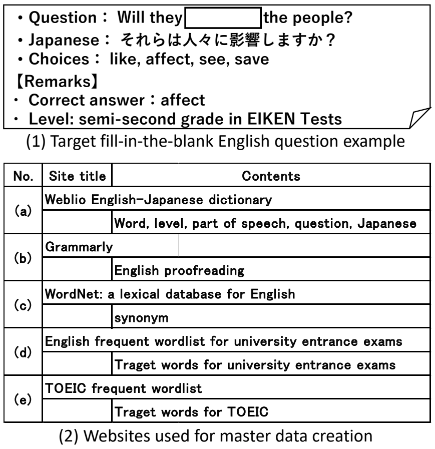


Fig. 1. Three-step master data creation method.



・Question： Will they ______ the people?
・Japanese： それらは人々に影響しますか？
・Choices： like, affect, see, save
【Remarks】
・ Correct answer：affect
・ Level: semi-second grade in EIKEN Tests

(1) Target fill-in-the-blank English question example

No.	Site title	Contents
(a)	Weblio English–Japanese dictionary	
		Word, level, part of speech, question, Japanese
(b)	Grammarly	
		English proofreading
(c)	WordNet: a lexical database for English	
		synonym
(d)	English frequent wordlist for university entrance exams	
		Traget words for university entrance exams
(e)	TOEIC frequent wordlist	
		Traget words for TOEIC

(2) Websites used for master data creation

Fig. 2. English question example and used websites.

the Japanese Article Numbering (JAN) codes. By providing ISBN and JAN codes for books, book shops can easily build product masters, including prices [8].

In the introduction phase of a business system, it has been mentioned that the preparation of the master is mandatory, and the system goal cannot be achieved unless the quality of the master is ensured [9]. However, business systems have been introduced in various fields, and databases, such as books and products, are not provided in all fields. Furthermore, the data of most websites that constitute big data is not always intended to be applied to individual business systems, and such data are scattered across various websites. Thus, it is difficult to create a master by directly utilizing big data.

One such area is English e-learning materials for Japanese, where English question masters with numerous questions must be created. To create these masters, a method using machine learning has been proposed [4]. In contrast, various translations and proofreading services, dictionaries, including example sentences, and a database of synonyms are provided on the cloud. So, it can be expected to utilize them for creating this master. Additionally, since words to be learned are recommended on various websites according to the purpose, such as the Test of English for International Communication (TOIC) level and university entrance exams, it is expected that questions suitable for each learner can be asked by utilizing such data in an integrated manner.

The motivation for this study is the question of whether big data can be used to efficiently and automatically create masters for such individual business systems. However, there are challenges. Firstly, once the master is created, it is not frequently updated. That is, for an individual business system, it is less cost-effective. Secondly, since various websites with diverse formats must be used as data sources, a mechanism for efficiently integrating data from those websites is required. And, I could not find the study that uses such various big data to automatically generate a master to be used in an actual business system and quantitatively evaluate its efficiency.

3. Proposal of three-step master data creation method

To address the issues described in Sec. 2, we propose a three-step master creation method for individual business systems from big data. Figure 1 shows the structure of the proposed method. First, data are acquired from each website. Then, the acquired data are integrated into the semi-structured common database. Finally, the database is used to create each master for multiple individual business systems to improve cost-effectiveness.

The target data are acquired from multiple websites on big data by scraping and stored in each collection of the individual database, as shown in Fig. 1 (1). Since this step involves the acquisition of data from websites with diverse data structures, collections store the data in a different format for each website.

The abovementioned collections are integrated by a common database via semi-structuring, as shown in Fig. 1 (2). Since this common database is used to create individual masters in the next step, it is constructed at a standardized level to some extent, such as the abovementioned book and product information databases. Using a semi-structured database, collections with various data structures can be integrated.

Necessary data are extracted from the common database and masters are created for each business system, as shown in Fig. 1 (3). This step corresponds to, for example, the process of creating a product master for each store from the product information database. In this step, by referring to various data, the target documents are extracted from the collection and some attribute values are determined. For product master, the best-selling products to be sold in the store are extracted and their prices are determined by utilizing the data of sales and price trends.

Transaction data, such as sales records for each product, is used as is input when business is performed using these masters. The transaction data are fed back and stored in the common database and used to improve the accuracy of recommendations and product purchases.

4. Implementation for English question master creation system

4.1. Target of English questions and big data

To evaluate the proposed three-step master creation method, we applied it to the master creation system for English fill-in-the-blank questions for Japanese learners. These questions are premised on use in e-learning and Japanese translations of sentences and answer choices are added, as shown in Fig. 2 (1). This e-learning provides comprehensive learning of words by preparing questions according to the learner's level.

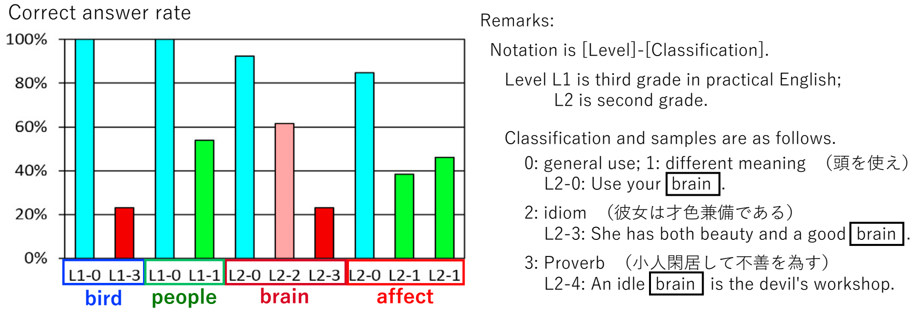
The first requirement for preparing such a question master is the need to prepare a large number of questions to learn words comprehensively, and it makes the workload to prepare the master enormously. Second, it is necessary to prepare questions according to the level and purpose of each learner. For example, questions on words that should be studied in high school are not suitable for junior high school students. Also, if the purpose of learning is based on TOEIC measures, it is effective to focus on the words that had appeared in previous TOEIC. Third, it is necessary to set the difficulty level (hereinafter, level) for each question. Though the level of a word is widely disclosed, one word often has multiple meanings or usages. So, the level of a question varies depending on the word's meanings used in the question. In other words, for the same word, there are questions with different levels.

For the first requirement, various English dictionaries and services are provided on the cloud, and by utilizing them a question master can be created. Figure 2 (2) shows websites used in this system. Figure 2 (2)(a) shows the website of the dictionary, from which words, their levels, and question sentences were acquired [7]. Figure 2 (2)(b) shows the website of the English proofreading service, which was used to check the acquired question sentences [5]. Figure 2 (2)(c) shows an English vocabulary database, which was used to exclude the correct word's synonyms from the choices in Fig. 2 (1) [3].

For the second requirement, since the level information of each word in the Test in Practical English Proficiency (EIKEN Tests) is indicated in the dictionary, as shown in Fig. 2 (1), questions could be classified according to school divisions, such as high school and junior high school. Furthermore, Figs. 2 (2)(d) and 2 (2)(e) are lists of frequent words in university entrance exams and TOIC, respectively, which were used to extract questions according to the learning objectives [6, 2].

For the third requirement, preliminary experiments in this study have shown that the word's level does not always match its question's level, as shown in Fig. 3. In this experiment, 13 university students answered questions in the form of Fig. 2 (1) for four words. In EIKEN tests, the level of two words is at the third grade and the level of the other two is at the second grade, and the meanings of the words are divided into four classes. Figure 3 shows the case of "brain", and their Japanese translation is shown in parentheses. For basic English sentences learned at school, the correct answer rate was about 85% or more; for other meanings and idioms, it was about 62% or less; for proverbs, it was about 23%.

Therefore, to extract appropriate questions according to the learner's purpose and level, the information of the websites mentioned in Figs. 2 (2)(a), 2 (2)(d) and 2 (2)(e) is insufficient. Thus, it is necessary to reflect the data of the correct answer rate for each question into the common database and utilize it to extract the questions for a suitable level. This situation corresponds to the "Feedback" of transaction data shown in Fig. 1.



Remarks:

Notation is [Level]-[Classification].

Level L1 is third grade in practical English;
L2 is second grade.

Classification and samples are as follows.

0: general use; 1: different meaning （頭を使え）
L2-0: Use your $\boxed{\text{brain}}$.

2: idiom （彼女は才色兼備である）
L2-3: She has both beauty and a good $\boxed{\text{brain}}$.

3: Proverb （小人閑居して不善を為す）
L2-4: An idle $\boxed{\text{brain}}$ is the devil's workshop.

Fig. 3. Assessment of inconsistencies between English words and question levels.

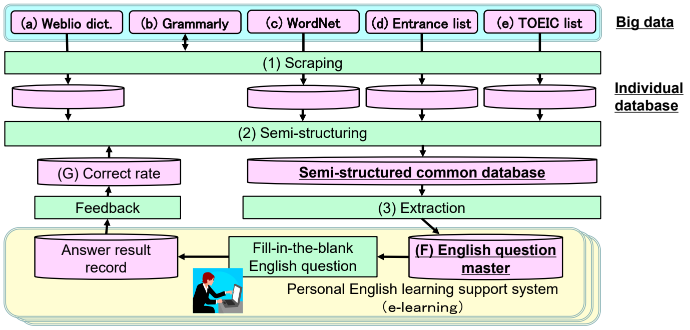
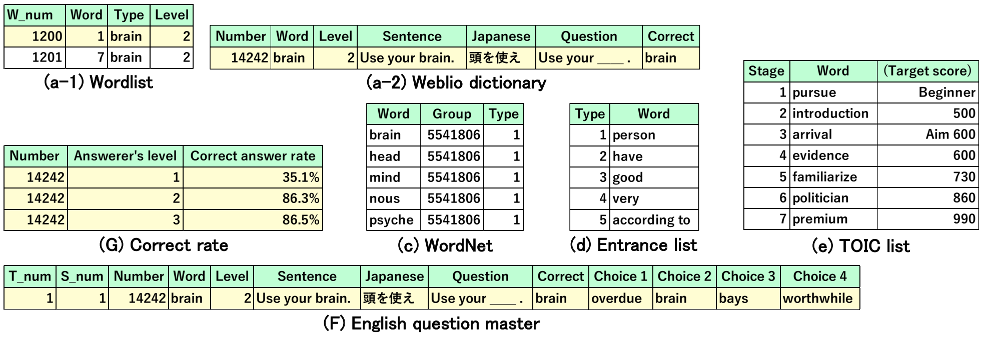


Fig. 4. Configuration of master creation system.



W_num	Word	Type	Level
1200	1	brain	2
1201	7	brain	2

(a–1) Wordlist

Number	Word	Level	Sentence	Japanese	Question	Correct
14242	brain	2	Use your brain.	頭を使え	Use your ____ .	brain

(a–2) Weblio dictionary

Stage	Word	(Target score)
1	pursue	Beginner
2	introduction	500
3	arrival	Aim 600
4	evidence	600
5	familiarize	730
6	politician	860
7	premium	990

(e) TOIC list

Number	Answerer's level	Correct answer rate
14242	1	35.1%
14242	2	86.3%
14242	3	86.5%

(G) Correct rate

Word	Group	Type
brain	5541806	1
head	5541806	1
mind	5541806	1
nous	5541806	1
psyche	5541806	1

(c) WordNet

Type	Word
1	person
2	have
3	good
4	very
5	according to

(d) Entrance list

T_num	S_num	Number	Word	Level	Sentence	Japanese	Question	Correct	Choice 1	Choice 2	Choice 3	Choice 4
1	1	14242	brain	2	Use your brain.	頭を使え	Use your ____ .	brain	overdue	brain	bays	worthwhile

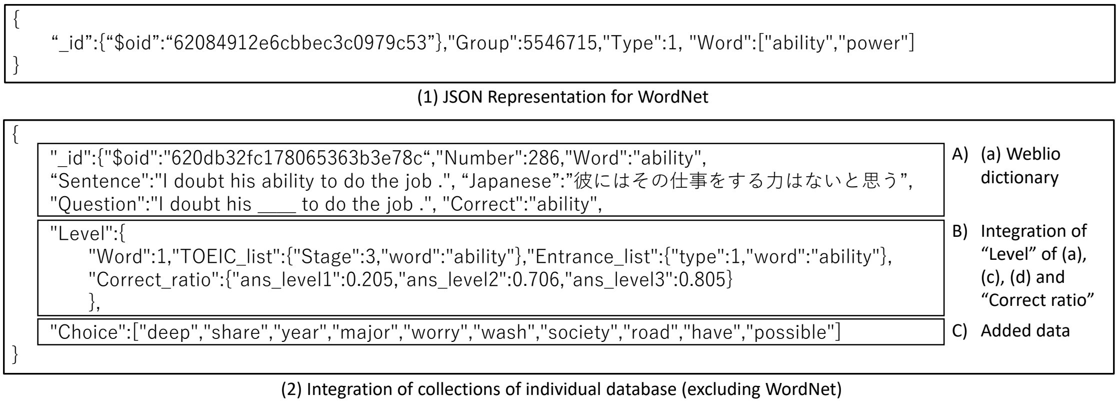
(F) English question master

Fig. 5. Example of individual data abstracts.

4.2. Configuration of English fill-in-the-blank master creation system

Figure 4 shows the configuration of the target English fill-in-the-blank question master creation system. The five websites displayed as big data are the same as those shown in Fig. 2 (2). The data of the correct answer rate for each question shown in Fig. 3 was utilized, and it is given as (G) Correct rate in Fig. 4.

Figure 5 shows examples of the main data abstracts. Figure 5 (a-1) shows a list of words acquired from Weblio English-Japanese dictionary, in which “Type” is the word’s parts of speech and 1 indicates a noun. Figure 5 (a-2) shows the set of attributes to create questions. Figure 5 (G) shows the correct answer rate, as mentioned above, in which “Answer’s level” indicates the level of the answerer in three stages: junior high school, high school, and university and beyond. Figure 5 (c) shows a case of synonyms, such as in the noun “brain”. In WordNet, synonyms



```
{
    “_id”:{“$oid”:“62084912e6cbbec3c0979c53”},"Group":5546715,"Type":1, "Word":["ability","power"]
}
```

(1) JSON Representation for WordNet

```
{
  "_id":{"$oid":"620db32fc178065363b3e78c“,"Number":286,"Word":"ability",
  “Sentence":"I doubt his ability to do the job .", “Japanese”:”彼にはその仕事をする力はないと思う”,
  "Question":"I doubt his ____ to do the job .", "Correct":"ability",
  "Level":{
      "Word":1,"TOEIC_list":{"Stage":3,"word":"ability"},"Entrance_list":{"type":1,"word":"ability"},
      "Correct_ratio":{"ans_level1":0.205,"ans_level2":0.706,"ans_level3":0.805}
      },
  "Choice":["deep","share","year","major","worry","wash","society","road","have","possible"]
}
```

A) (a) Weblio dictionary

B) Integration of “Level” of (a), (c), (d) and “Correct ratio”

C) Added data

(2) Integration of collections of individual database (excluding WordNet)

Fig. 6. Semi-structured common database structure.

are classified by “Group”. Thus, it is necessary to exclude such synonyms from the choices in question. Figure 5 (d) shows the words that frequently appear in university entrance exams by part of speech (Type). Figure 5 (e) shows the TOEIC target levels, and the respective target scores are noted in the far-right column “(Target score)”.

From these data, the English question master in Fig. 5 (F) is created, with four choices, including the correct answer corresponding to the underlined part in the question. So, these data need to be integrated to extract the questions according to the learner’s level, and the integrated data were implemented as a semi-structured common database shown in JSON. Figure 6 (1) shows the implementation of the document for storing WordNet data, and words of the same group are stored as an array. Figure 6 (2) shows the data other than WordNet. Figure 6 (2) A) shows the data of Fig. 5 (a-2) other than “Level”; Fig. 6 (2) B) shows the data about level, including the level of Fig. 5 (a-2); Fig. 6 (2) C) shows the additional choices consists of 10 choices for standardization.

4.3. Implementation of each processing step

The system was implemented on a Windows 10 PC, using MongoDB Ver. 5.0.2 as the database, Python Ver. 3.7.10 as the program, and pymongo Ver. 3.12.0 as the MongoDB driver. For accessing big data and cloud services, we used Google Chrome Ver. 95.0.4638.69 as a browser, Selenium Ver. 3.141.0 and Chrome Driver as programs. Grammarly was used as an add-on for Google Chrome, and the results were obtained by entering English sentences into Google Translate.

In the scraping of Fig. 4 (1), data were acquired by scraping, except WordNet which is provided as a CSV file to be downloaded. Particularly, the Weblio Dictionary has a complicated structure, so its data was acquired by the following procedure. First, the words shown in Fig. 5 (a-1) were collectively acquired from the alphabet index. Then, the necessary data for creating questions shown in Fig. 5 (a-2) were acquired from each word page. To prevent the increase in loads to the website, a one-second wait was added after accessing each word page. All example sentences of each word are described on one page with word information. Acquired sentences were checked by Grammarly, and a sentence was excluded if an error was detected in it. This procedure aims at both excluding unsuitable sentences for questions and verifying the scraping process. Since Grammarly’s proofreading function is provided as a cloud service, 9 seconds wait in total was added for each sentence to wait for a response.

In the semi-structuring of Fig. 4 (2), the common database was created from the individual databases. The levels in Fig. 6 (2) B) was added to those in Fig. 6 (2) A), namely, Weblio Dictionary. Then, Choice in Fig. 6 (2) C) was added by the following procedure. First, for each question in Fig. 5 (a-2), its word’s level and type were acquired from the Wordlist in Fig. 5 (a-1). Then, the words below this level and the same type were acquired from Fig. 5 (a-1); synonyms of this word were acquired from WordNet data shown in Fig. 6 (1). Finally, an array of 10 choices, excluding synonyms from the acquired words, was added.

From this collection, the question master shown in Fig. 5 (F) was created according to the specified number of questions and choices. Since MongoDB did not provide a random document extraction function, the keys of the

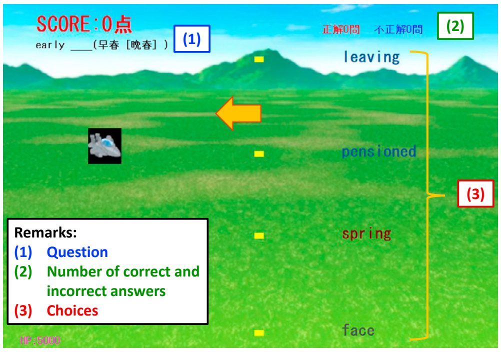


Fig. 7. Implementation of English e-learning.

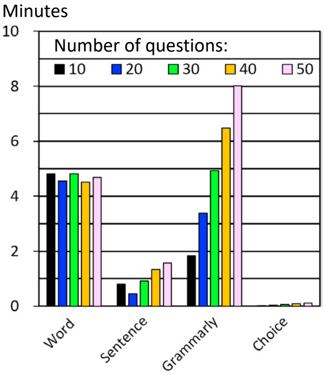


Fig. 8. Elapsed time of each process to acquire data and add choices.

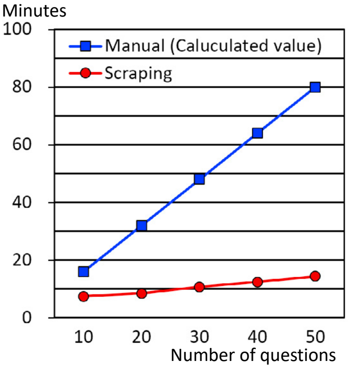


Fig. 9. Comparison of elapsed times by manual and system.

document (_id) that met the specified conditions were retrieved first. Then, their array was created and shuffled, and a subarray of them was selected. Lastly, the documents for the question master were acquired collectively under the condition that their keys were included in the selected subarray. Three choices were randomly selected in each document and shuffled with adding the correct answer to create the choices of the master.

The e-learning question was implemented as a shooting game, as shown in Fig. 7. This implementation aimed at concentrating on learning and improving the motivation to learn by the game. A question is presented in the upper left of the screen, and its choices flow in the direction of the arrow from the right. By shooting the correct answer, the score increases; the wrong answer decreases the point, which is shown by “HP” at the lower left end of the screen, and the game is over when the point reaches zero. HSP Ver. 3.6 was used to implement this game.

5. Evaluation of master data creation efficiency

The proposed method was evaluated by comparing the efficiency of automated scraping and manual acquisition, as well as MongoDB and MySQL. The PC used for scraping was equipped with an AMD Ryzen 5 3500 3.6 GHz CPU, Geforce GTX 1650 SUPER GPU, and 16 GB of memory. The PC used for database access was equipped with Intel Core i9-10850k 3.6 GHz, GeForce RTX 3090 GPU, and 64 GB of memory.

5.1. Evaluation of website data acquisition efficiency by scraping

In the scraping evaluation, WordNet data was created in advance as a collection of Fig. 6 (1). The number of choices was 3, except for the correct answer. The level data was not used. Figure 8 shows the elapsed time of each process to acquire terms in Figs. 5 (a-1) and 5 (a-2) and to add choices. “Word” shows the time taken to acquire the words in Fig. 5 (a-1), and the number of acquired words was 12,326. “Sentence” shows the time taken to acquire the attributes in Fig. 5 (a-2); “Grammarly” shows the time taken to proofread English using Grammarly. “Choice” shows the time to add choices as mentioned in Sec. 4.3 and creates the correction shown in Fig. 6 (2). In “Word”, since all words are acquired, the time taken is approximately constant. In other cases, the time increased as the number of questions increased. Since “Choice” was performed only by database operations, the time is so short because there is no waiting time due to scraping shown in Sec. 4.3.

Fig. 9 shows the comparison between the elapsed time and the manual acquisition time. In the manual acquisition, the procedure was the same as in Fig. 8 except for the processing shown in “Word”, and the results were written on MS Excel, not the database. By measuring the time to create 10 questions, the average time per question was 90 seconds, so the graph of “Manual” in Fig. 9 was drawn based on this time. In the scraping evaluation, the first 10 questions took 7.5 minutes for acquisition, then the subsequent 10 questions increased by 1.7 minutes, i.e., 10.4 seconds per question. In other words, master creation by automatic scraping was 8.7 times more efficient than manual acquisition

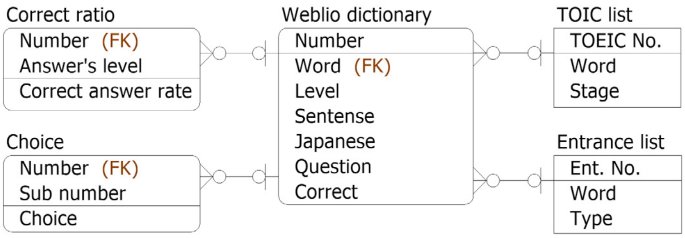


Fig. 10. ER diagram in with MySQL.

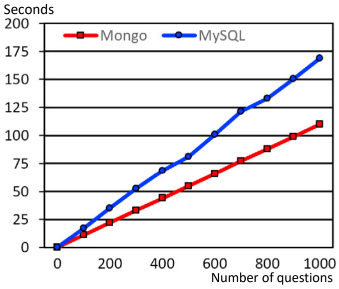


Fig. 11. Comparative evaluation of synonym retrieve efficiency.

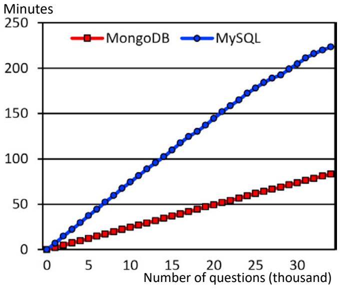
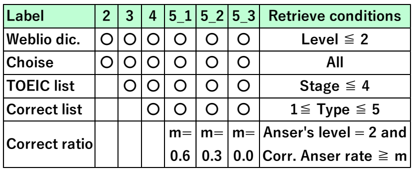


Fig. 12. Comparative evaluation of Choices creation efficiency.



Label	2	3	4	5_1	5_2	5_3	Retrieve conditions
Weblio dic.	○	○	○	○	○	○	Level ≦ 2
Choise	○	○	○	○	○	○	All
TOEIC list		○	○	○	○	○	Stage ≦ 4
Correct list			○	○	○	○	1≦ Type ≦ 5
Correct ratio				m= 0.6	m= 0.3	m= 0.0	Anser's level = 2 and Corr. Anser rate ≧ m

Fig. 13. Common table retrieve conditions.

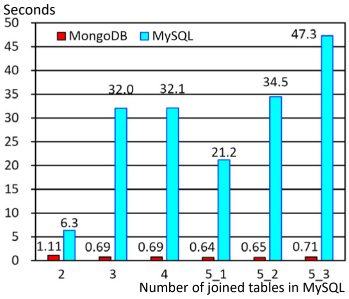


Fig. 14. Comparative evaluation of 5,500 question master extraction efficiency.

when a lot of questions were created. Furthermore, since everything could be executed automatically, there was no burden on the worker.

5.2. *Efficiency evaluation of semi-structured database*

To evaluate the efficiency of the common database implemented by the semi-structured database, comparative evaluations were conducted between MongoDB and MySQL for semi-structuring and extraction, as shown in Fig.4. To avoid the sparse problem and to be able to add arbitrary websites at any time, a normalized structure of MySQL was used, as shown in Fig. 10.

First, for WordNet collections, the efficiency of synonym retrieval was evaluated. In MySQL, the data was saved in the format of Fig 5 (c); in MongoDB, it was saved in an array of each document, as shown in Fig. 6 (1). The number of data was 313,620; the number of groups was 53,525, which implies that the average number of synonyms was 5.9. Using the array of randomly extracted words, the time to create arrays of synonyms for each word was measured for every 100 words up to 1,000 words. On a simple average of times, MongoDB was 1.54 times more efficient than MySQL, as shown in Fig. 11,

Second, the time to create the common database using the procedure mentioned in Sec. 4.3 was evaluated, in which choices shown in Fig. 6 C) were added, but the addition of level data was excluded. For MongoDB, the choices were added as an array; for MySQL, the choice data were inserted into the Choice collection shown in Fig. 10. Figure 12 shows the results. The number of questions was 34,548, of which the transition of elapsed time is shown for every 1,000 up to 34,000. The average elapsed time to add 1,000 questions was 146.93 seconds for MongoDB and 394.26 seconds for MySQL, respectively, i.e., MongoDB was 2.68 times more efficient than MySQL.

Third, the efficiency of retrieving questions from the common database with the designated conditions and storing them into the collection of question masters was evaluated. Its procedure was as mentioned in Sec 4.3, and the number of questions was 5,500; the number of choices was three, except the correct one. Figure 13 shows the retrieval

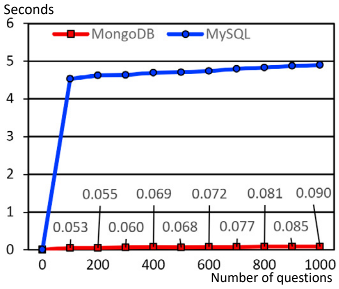


Fig. 15. Transition of question master extraction elapsed time (case: 5_3).

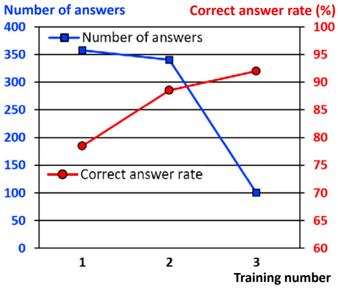


Fig. 16. Evaluating the effectiveness of question mastery in e-learning.

conditions. Figure 14 shows the results. The row of "Label" in Fig. 13 corresponds to the horizontal axis of Fig. 14, and it indicates the number of join tables in MySQL. For the case of Label 5, the target correct answer rate decreases as the right-side number increases, as indicated by "m", so the target questions increase.

In MySQL, the efficiency tended to deteriorate significantly as the number of tables to be joined increased, but the efficiency also depended on the number of targets, as shown in the case of Label 5. In contrast, in MongoDB, the efficiency did not deteriorate greatly depending on the conditions. For example, in the case of 5_3, the efficiency of MongoDB increased 66.7 times compared to MySQL.

Figure 15 shows the results of measuring the elapsed time transition for extracting the question master for every 100 questions up to 1,000 in the case of Label 5_3. For both MySQL and MongoDB, the first 100 questions took most of the time, and the rate of increase was reduced afterwards. The time taken for the first 100 questions was 4.5 seconds in MySQL, and it shows that the join operation of tables is a heavy load.

5.3. *Evaluation of effectiveness in e-learning*

To evaluate the effectiveness of the English questions created using the proposed method, the questions were used in actual e-learning. The number of questions that were repeatedly and randomly asked was 100. Figure 16 shows the evaluation results of the number of total answers and correct answer rate in the case where a learner played the game three times. The learner was a game developer and was proficient in the game. The learner could continue until the game was over by selecting designated incorrect answers.

The correct answer rate improved along with a repeat of the game, which was almost the same tendency for all other learners. Therefore, it is considered that the questions created using this method were effective in actual e-learning. In the third time of the training number, the number of total answers decreased extremely. It is considered that the decrease in the number of answers is due to a decrease in the learners concentration because the learner repeated the questions without a break.

6. Discussion

The most time-consuming process was scraping, which was due to avoiding the load on websites, as shown in Fig. 8 and MongoDB's graphs in Figs. 14 and 15. On the contrary, the question master extraction time was small: it was about one second for 5,500 cases. So, it is considered that the most effective method of scraping is to build a common database covering the masters required for each business system firstly and then extracts each master individually.

For the problem that it takes time to scrape and integrate into the common database, the efficiency of the system can be improved by distributed processing. For example, the "Sentence" and "Grammarly" in Fig. 8 used different websites, so they can be accessed in parallel. Additionally, for the integration into the common database, by dividing the data into question groups and integrating them after processing, it is considered that performance could be improved. The evaluation of performance improvement by such decentralization is a challenge for future study.

The data from each website on big data has individual attributes and structure, as shown in Fig. 5. To use a relational database, the sparse problem must be avoided and the additional websites must be accommodated. However, the join operation caused significant efficiency degradation, as shown in Fig. 14. In contrast, since MongoDB stores all the data in a single collection, there is no deterioration. As a result, for retrieving a single collection, MongoDB was 1.54 times more efficient than MySQL, as shown in Fig. 11. It expanded 66.7 times for the five-collection join shown in Label 5_3 of Fig. 14. Furthermore, in the comparative evaluation of collection creation for choices, MongoDB was 2.68 times more efficient than MySQL, as shown in Fig. 12.

In summary, MongoDB, a semi-structured database, is considered efficient for both database creation and retrieval, in which various websites data on big data are integrated.

Since there are various business systems, the composition and operation of the master also vary widely. The validity of the proposed method for masters in other fields remains for future studies.

7. Conclusion

To introduce a business system, the cost of preparing its master is often a major issue. Although various data are released as big data, they are not always consistent with the target master. Given that the data are distributed to multiple and various websites, it is difficult to utilize big data for the master preparation directly.

In this study, we proposed a three-step master creation method, in which necessary websites' data are stored in a semi-structured common database, and then the master of each business system is extracted from it. This method was evaluated by an English question master creation system. The results showed that the proposed method is effective: the cost of master extraction from the common database was small compared to that of its creation; by a semi-structured database, it became possible to create a common database and extract master data from it efficiently.

Future issues include the performance improvement evaluations by decentralizing the scraping and the creation of the common database, as well as the validity of the proposed method in other fields.

Acknowledgements

This work was supported by JSPS KAKENHI Grant Number 19K11985.

References

[1] Ang K. L. -M., Ge F. L., and Seng K. P. "Big Educational Data & Analytics: Survey, Architecture and Challenges," *IEEE Access* **8**: 116392–116414.

[2] Eigo Love, "English words list for university entrance exams." https://www.eigo-love.jp/eitango/#list-center (referred March 8, 2022).

[3] George A. M. (1995) "WordNet: A Lexical Database for English," *Communications of the ACM* **38** (11): 39–41, https://wordnet.princeton.edu/ (referred March 8, 2022).

[4] Goto, T., Kojiri, T., Watanabe, T., Iwata, T., and Yamada, T. (2010). "Automatic generation system of multiple-choice cloze questions and its evaluation." *Knowledge Management & E-Learning: An International Journal* **2** (3): 210–224.

[5] Grammarly Inc. "Grammarly," https://www.grammarly.com/ (referred March 8, 2022).

[6] English banana, "TOEIC Frequent Word List." https://englishlevelup.net/toeic-word-list/ (referred January 8, 2022).

[7] GRAS group. "Weblio English-Japanese / Japanese-English dictionary." https://ejje.weblio.jp/ (referred March 8, 2022).

[8] GS1 Japan (2021) "GS1 Japan Handbook 2021-2022" https://www.gs1jp.org/assets/img/pdf/GS1-Handbook_2021-2022.pdf (referred March 8, 2022).

[9] Haug, A., and Arlbjørn, J. S. (2011) "Barriers to master data quality." *J. Enterprise Information Management* **24** (3): 288–303.

[10] International ISBN Agency "isbn." https://www.isbn-international.org/ (referred March 8, 2022).

[11] Jose, B., and Abraham, S. (2020) "Performance analysis of NoSQL and relational databases with MongoDB and MySQL." *Materials today: PROCEEDINGS* **24**: 2036–2043.

[12] Kaur, K., and Rani, R. (2013) "Modeling and querying data in NoSQL databases." *2013 IEEE international conference on big data*: 1–7.

[13] Moniruzzaman, A. B. M., and Hossain, S. A. (2013). Nosql database: New era of databases for big data analytics-classification, characteristics and comparison. *Int. J. Database Theory and Application* **6** (4): 1–13.

[14] Pham, Q. V., Nguyen, D. C., Huynh-The, T., Hwang, W. J., and Pathirana, P. N. (2020) "Artificial intelligence (AI) and big data for coronavirus (COVID-19) pandemic: A survey on the state-of-the-arts." *IEEE Access* **8**: 130820–130839.

[15] Wang, J., Yang, Y., Wang, T., Sherratt, R. S., and Zhang, J. (2020) "Big data service architecture: a survey." *J. Internet Technology* **21** (2): 393–405.