Semantic-Based Search Engine System for Graph Images in Academic Literatures by Use of Semantic Relationships

Sarunya Kanjanawattana and Masaomi Kimura

***Abstract*—Information retrieval is a baseline of search engine systems. There is a very large amount of data published on the Internet that cannot be manually searched. However, search engine systems should not only present relevant results but also obtain new knowledge from the user's searches. For example, new knowledge in academic research areas may be presented in graph images. In this study, we utilize methods to extract graphical and textual information from graph images and store this new knowledge in an ontology. We propose a search engine system that is applicable to an ontology that contains this extractable information, which is extracted from images with graphs. The developed ontology is useful because users can acquire a considerable amount of knowledge that is discovered from the semantic relations in the ontology. To evaluate the search engine system, ten participants tested the system and responded to their feedback. The results indicate that the proposed system provides accurate and relevant results; moreover, as indicated by the higher F-measure values comparing to an Elasticsearch-based search engine system, the performance of our system is highly acceptable. It clarified that the ontology-based search engine system provides precise and concise information outperforming than the Elasticsearch-based search engine system.**

***Index Terms*—Ontology, search engine system, graph information, semantic relations, information retrieval.**

I. Introduction

An ontology specifies the representation of a conceptualization. Recently, ontologies have been recognized as an important feature of information retrieval systems owing to their ability to link knowledge in different areas of the ontology. Ontology-based search engine systems often acquire more useful information than traditional search engine systems because users can find not only particular concepts obtained by a given query but also other related concepts.

However, for the practical use in ontologies, an application should be developed in order to present understandable results from an ontology query to users because it is extremely difficult for the average end user to realize the results from the ontology query. Typically, to input a query via an ontology, it is necessary for the user to be skilled in a query language; in addition, a specific ontology realization will be required. This creates problems for a general user who would then avoid using the complex system. Therefore, we introduce a handy and capable search system that does not require any computer skills by constructing a web-based application with user interfaces.

In recent years, there has been a substantial amount of research on information retrieval [1]. There are many types of data that are regular targets for searching systems, including text [2] and image [3] data. In particular, an image-based information retrieval is a growing topic in several study fields (e.g., computer vision and knowledge-based information retrieval) because methods for extracting data from images are more complicated than those for extracting data from text. Hence, researchers require particular methods to extract image information effectively. For example, a system of image content extraction could analyze an image's low-level features [4] such as colors.

Images in the academic literature containing essential information may not be described in any other part of the paper. Moreover, graphs in academic literature always present statistical data, relations, or comparisons that are different to a generic image such as a photo. Indeed, it is very useful for users if they can acquire information from a graph and thus obtain new knowledge. Hence, not only the textual information but also the graphical content should be utilized for queries in search engine systems. However, dealing with both the graphical content and the linguistic information in the graph can result in a semantic gap problem, which characterizes the difference between linguistic and graphical representations. If the gap is large, information extracted from the graphical and linguistic representations may be misunderstood. Therefore, one solution is to use ontologies or semantic relations to narrow the gap.

In the past, we have attempted to extract graphical information from a graph, including graph components such as axis titles and legends [5] and graph data information such as data located in the axes of the graph [6], [7]. We constructed an ontology and stored the information extracted from the graphs and their descriptive contents (i.e., captions and cited paragraphs) in the ontology. Objectives of the proposed system are to utilize an ontology based on the structure designed in our previous study [6] and to propose an ontology-based search engine system.

In the past several decades, there have been several studies related to semantic search engine systems [8]–[10]. Li *et al.* [11] developed a fuzzy search by allowing mismatches between query keywords and answers. To explore data freely, this fuzzy search accepted keywords

Manuscript received August 15, 2019; revised October 7, 2019.

Sarunya Kanjanawattana is with Institute of Engineering, Suranaree University of Technology, Nakhonratchasima, Thailand (e-mail: sarunya.k@sut.ac.th).

Masaomi Kimura is with the Department of Information Science, Shibaura Institute of Technology, Tokyo, Japan (e-mail: masaomi@sic.shibaura-it.ac.jp).

doi: 10.18178/ijmlc.2019.9.6.880

even in the presence of minor errors. Based on this existing study, we realized that results or knowledge provided by a conventional system are limited because they are solely dependent on given keywords. In the most recent decades, studies on information retrieval have advanced to semantic systems utilizing ontology concepts that enhance and extend the obtained knowledge based on user specifications. Jayalakshmi *et al.* [12] proposed a semantic search engine system that depended on inverse document frequency and text mining. The proposed search engine system created the search indexing using the contents of the files to retrieve the relevant document from a computer. Another existing study constructed a scalable semantic search for geospatial data [13] in which an application layer and a search service that provided a specific search functionality inspired by resource description frameworks were introduced.

In this study, we propose an ontology-based search engine system that utilizes the ontologically stored textual and graphical information that is extracted from images containing graphs, including their captions and paragraphs. Note that, in this study, we used graph images in academics as a dataset. The graph is a diagrammatic illustration of a set of data. This system is necessary for researchers and students who need to read many kinds of literature and analyze graph images. In our proposed system, users can select questions and assign required settings to complete their queries. Depending on the selected questions, the search engine system provides relevant graph images and extended information.

II. Related Works

Searching useful information is a broad central area in information retrieval and knowledge acquisition. Many applications have been active currently, such as Google [14]. Hearst *et al.* [15] developed a search engine that provided a way to access biological scientific literature. They used the Lucene open source search engine to index, retrieve, and rank the text. Definitely, everyone admits that they are very useful and influent their daily life because data now are online and can be easily searched on the world wide web. Although regular search engine offers much information based on the use of keywords, the ontology-based search engine can provide more relative knowledge due to its semantic structure that improves search precision. Gauch *et al.* [16] introduced an ontology-based method to suggest information navigation using a user profile structured as a weighted concept hierarchy. Their system automatically created user profiles reflecting the user's interests that produce moderate improvements of search results.

Furthermore, not only text [17], [18] but also images can be searched by using ontology. Semantic web, ontology and image information extraction offer a new way to annotate and retrieve image data [19]. As presented in [20], Hyvonen *et al.* considered several situations when users were encountered complicated and semantical images and knew how ontology can be used to realize them. To prove their concept, they implemented a system for image annotation depended on ontology and the same conceptualization. Finally, their system provided a recommendation of semantically related images to users. However, most systems that deal images often face the problem of the semantic gap.

Capturing image semantics opens up a new field of study by integrating multi-disciplines to overcome the existing problems [21], [22]. A typical solution to minimize the gap is to utilize both graphical and textual information in order to obtain relevant knowledge. Zhao *et al.* [23] proposed a method to extract the underlying semantic structure of web documents by latent semantic indexing for textual information to cluster co-occurring keywords or concepts. Users used a particular keyword to retrieve documents that may not include the keyword but contain other keywords in the same cluster. For graphical content, they extracted low-level image features using color histograms and color anglograms. Chen *et al.* [24] developed a vertical image search engine integrating both textual and visual features to improve retrieval performance. To bridge the semantic gap, they captured the meaning of each text term in the visual feature space and repeatedly measured the weight of visual features according to their significance to the query terms. Moreover, they considered user intention gap that can infer visual meanings behind the textual queries. The previous studies above conducted experiments and their results showed the improvement of precision and recall.

III. Methodology

We utilize graph component extraction [5], graph information extraction [6], and optical-character-recognition-error (OCR-error) correction [25] to obtain the graph information. This graph information is then added to an ontology with the structure designed in the previous study [6]. Note that the size of a graph image dataset is 636 images containing two graph types: bar graph and two-dimensional chart (2Dchart). The 2Dchart represents a line graph and plot graph. We merge them because their structures are similar the present continuous data on both X- and Y-axis; whereas, the bar graph demonstrates a discrete data on X-axis. A graph structure used in this study displays in a form of a two-dimensional axes graph because it mostly finds in academic literature.

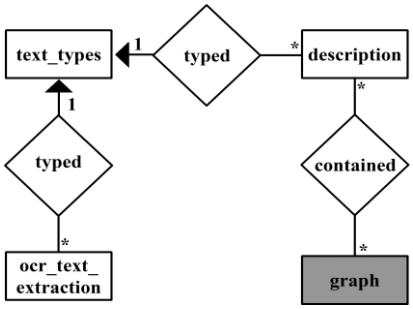

Fig. 1. A part of a database storing generic data related to the graph images.

A. Database Design

We used a relational database to store data related to our target data (i.e., a collection of graph images, including their captions and text paragraphs) and other necessary information such as captions, text paragraphs, and graph profiles. It was constructed due to two major purposes: to store the graph information (e.g., captions, paragraphs, and

graph profiles) and to record user evaluation feedbacks. The graph information was an important data used to create our ontology. Moreover, due to an evaluation purpose, the system allowed the users to comment and validate results obtained from queries accessing to both search engine systems. Those feedbacks were recorded for statistical analysis.

Five tables were used to record the following information from each graph: graph, contained, description, text_types, and ocr_text_extraction (Fig. 1). The "graph" table collected the profiles of graph images, such as the graph name. The "description" table contained the graph's captions and the paragraphs that referenced the graphs. The "text_types" table contained the different types of the graph's descriptions (e.g., caption, X-title, and legend). To acquire the graph components, we used OCR to first recognize and convert them into digital data. These data were stored in the "ocr_text_extraction" table.

Regarding the rest of the tables, they were utilized to record user feedbacks (Fig. 2). We collected not only the user evaluation feedbacks but also the results from queries. User table collected the user information, such as name and major. Question, Condition, Feature tables primarily kept the inquiry details, for example, questions used for a query. Query and Option tables stored such information of user's query on each iteration. After the users inquired queries to the system, some relevant results should be returned to the search engine systems, and those were evaluated by the users. Then, the obtained results and user evaluation were collected into Query_result_relevance table.

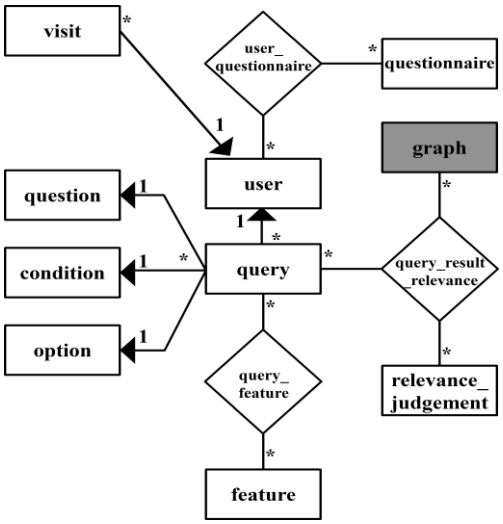

Fig. 2. A part of a database storing feedback data for the search engine system in Feedback mode.

B. Ontology Design

The ontology used in this study was based on the structure design in [6], but we redesigned it to be more applicable and to meet the requirements of the search engine system analyzed in this study. (Fig. 3).

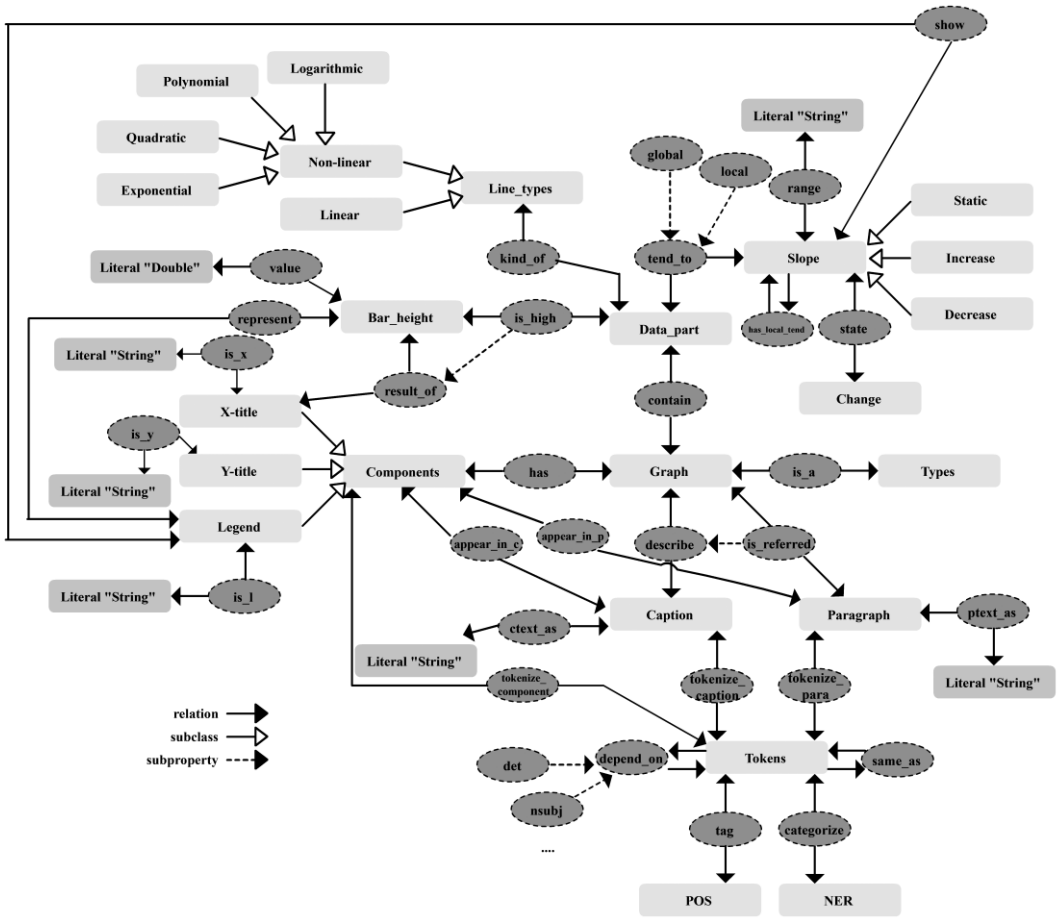

Fig. 3. Ontology design with new relations.

The ontology had been created to support our smart search engine system. It contained 24 entities and several object properties. Here, we would explain some important concepts to globally realize the ontology. The core concept

of the ontology should be "graph" entity representing the concept of the graph in our data set. The "graph" itself certainly includes much knowledge and information that can be used in the search engine system. A graph always has graphical data, e.g. lines, numbers, and bars that were represented in the "Data_part" entity relating to the concepts describing statistical data in a graph. The three components, i.e. X-title, Y-title, and legend, inherit from the "Components" concept. We extracted the necessary components from the graph because we realized that knowledge sometimes appears in graph components' relationship. Therefore, to capture the most significant information, these components are also required. Moreover, all graphs here were selected in literature; then, captions and mentions in paragraphs surely reside nearby the graphs. These graph descriptions explain the insight of the graphs and clarify their data by displaying some graphics. To assist these features, this ontology includes the concept of caption and paragraph. Text description was tokenized into tokens that were also represented as the "Token" entity.

Herein, we describe the updated parts that differ from the previous version of the ontology. Note that the previous ontology was simply applicable to singular data in the plot, line, and bar graphs but could not handle multiple data labels. Thus, we have since added a few relations that allow the ontology to be applicable to bar graphs containing multiple data labeled in a legend. A relation named "show" now connects the Legend and Slope classes because it describes the data tendency of each data label that belongs to different categories in the X-title class. A "represent" relation indicates the height of each data.

C. System Implementation

An ontology-based search engine is a search engine application that utilizes ontologies to fulfill user inquiries. With ontologies, it is possible to obtain relevant results to a query as well as to obtain new extended knowledge. Such a search engine system helps users to retrieve images of graphs that contain information they require, such as a relation between the X-and Y-axis labels and a comparison of each bar in a bar graph. The users can easily comprehend the graph and its descriptive details because the search engine system precisely provides detailed information such as main ideas and tendencies. The system allows users to select specific questions for inquiring; moreover, some settings must be accepted to restrict the amount of obtained results. Fig. 4 displays the overall process of this system implementation.

We implemented the system containing two different modes: a search mode and a feedback mode. The search mode was used to search and inquire to the system by inserting some keywords and specific questions; then, relevant results were returned that were displayed in a search page. Whilst, the feedback mode contained a particular feature used for acquiring the user's feedback. This mode was proposed for an evaluation purpose. After the results were retrieved, the user would analyze and decide them as either relevance or irrelevance based on their intention. They should evaluate every result presenting on both our system and a traditional search engine, which is called Elasticsearch (ES). In addition, in the feedback mode, we required user profiles filled on a user page) and evaluation opinions filled in a questionnaire page.

We implemented the described search engine system in a search application. The developed application can query the search engine system by specifying some keywords and specific questions. The relevant results are then returned and displayed in a search page. This system was designed to support simplicity and immediate availability. To that end, only necessary functions such as the query settings are shown on the web page. Three sections such as a menu section, the inquiry section, and results section are presented on the main search page, as shown in Fig. 5.

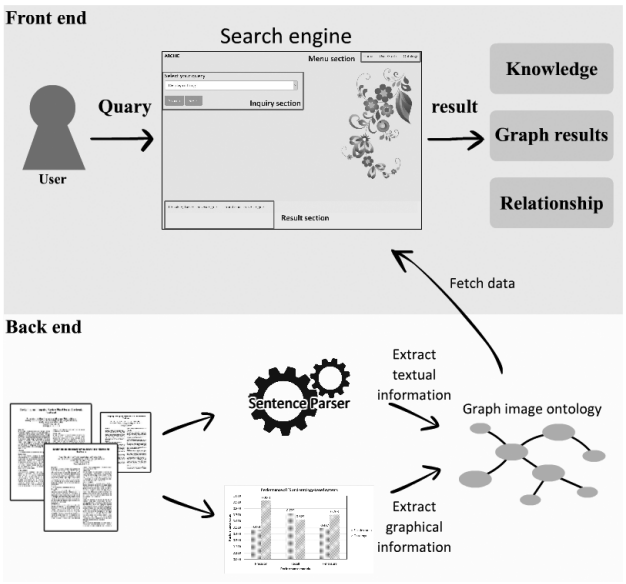

Fig. 4. A process of system implementation.

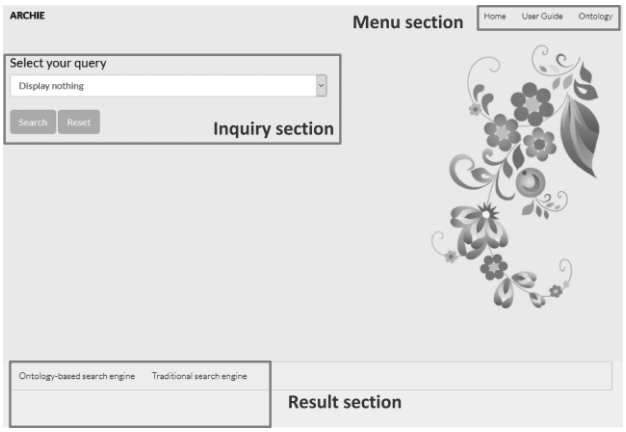

Fig. 5. Illustrating a user interface of a search mode with three sections.

The menu section contains three tabs, namely home, user guide, and ontology. The home tab is the default screen when the system is launched in the search mode. The user guide is a page that briefly explains the system and its components, including a guideline of the system process and examples of system simulations. The ontology tab displays the ontology schema that is used in this system.

In the inquiry section, users can select questions and input some required settings. The acquired relevant results are then displayed on in the results section. In addition to the question option could that can be selected by the users, we offered a few options that can help the users to filter unnecessary results, (i.e., conditions and features.

The condition box contains five conditions. First, the

users can restrict the graph type. We distinguish the graph types into two types such as bar graphs and 2Dcharts. Second, the users can select results that belong to a specific group. Third, the results shown in the results section can be filtered based on a specific regression type. (For example, a user might need only graphs that have linear regression.) Fourth, the users can select the results with a specific tendency such as increasing or decreasing. Finally, for the line and plot graphs, a local tendency is also a significant option, because changes in the graph might identify essential information. Thus, users can filter the results based on the data variation. The feature box was created to cover the needs of all users because a user might require additional information such as the graph caption or X- and Y-labels.

In the results section, results from our ontology-based and ES-based search engine system are presented. Depending on the user's system, a user can independently choose a tab to examine the results.

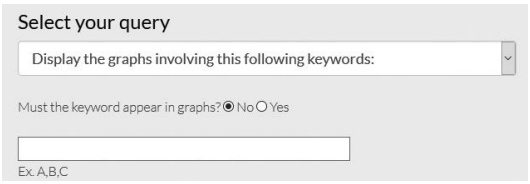

Fig. 6. A query form of question 1.

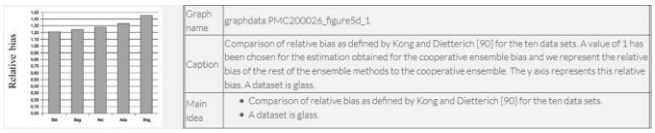

Fig. 7. Example results for question 2 that presents the main idea extracted from a caption.

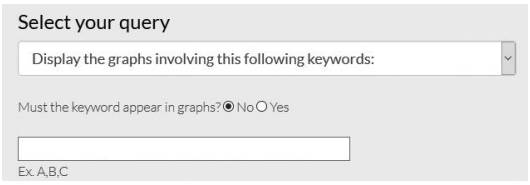

Fig. 6. A query form of question 1.

Herein, we discuss the questions that are included in the system and the settings that must be entered by the user. There are six queried questions, described as follows:

- Question 1: Display the graphs involving the following keyword(s).
- Question 2: Display the graphs involving the following keywords and their main idea of captions.
- Question 3: Display the graphs involving the following keywords and their maximum and minimum values of graphs.
- Question 4: Display the graphs relationship extracted from axis titles.
- Question 5: Display the relationships between two different tokens.
- Question 6: Display the comparison of bar values on different X-categories but same data label.

The first question is the most basic because it is similar to a keyword-based search engine system (e.g., Google search). A few settings are required and must be completed by the user. For example, a user may need graphs that feature a specific inputted token in the graph for deep discussion on a particular topic. An example query form for Question 1 is illustrated in Fig. 6. The user simply inputs at least one keyword to the text box separated by commas (for example, the string "data, test, accuracy" can be inputted to the text box). Moreover, the user can specify whether the keyword(s) must appear in the graph's components (e.g., X-label, Y-label, or legend) by choosing either the "yes" or the "no" radio button above the text box. An optional text box uses to asks the user's intention for the query; however, to complete the evaluation, the user should describe their intention for their query. For example, a user may input keywords such as "neural network, accuracy, image," and the intention would be to obtain graph images relating the accuracy of neural networks when dealing with images.

The second question requires only keyword(s) from the users to produce the relevant results. Moreover, the question asks for the main idea of the graph descriptions (e.g., the caption). Therefore, an extra feature has been added to the results section (Fig. 7), that presents the main idea. Sentences containing the main idea are selected by analyzing the appearance of keywords and the first sentence of the paragraph. A user can use this question to summarize information to realize the underlying concept of a graph.

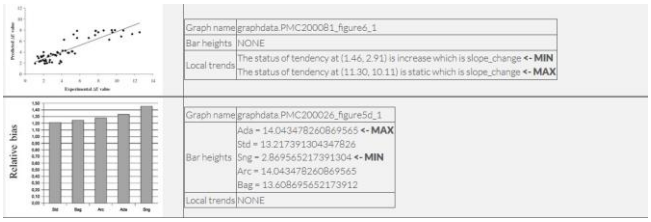

Fig. 8. Example results for question 3 presenting which data represents the maximum and minimum values comparing to other corresponding data.

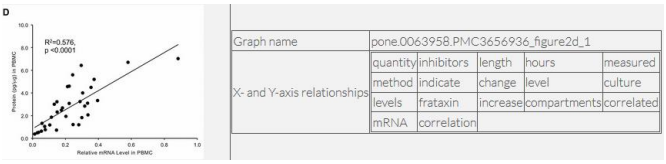

Fig. 9. Example results for question 4 that displays discovered relationships existed between graph axes.

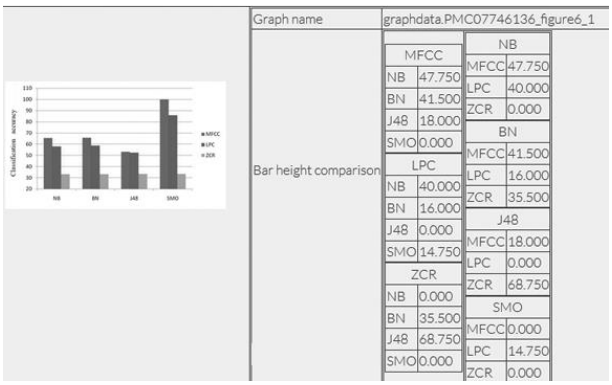

Fig. 10. Example result for question 6 presenting the bar height of among categories with two different viewpoints.

The third question is similar to the second question, and it requires keyword(s) to be set. The bar height and local trend features are initially selected and displayed in the results section, including the highest and lowest values identified (Fig. 8). However, for a bar graph containing multiple data, it is difficult to identify which the highest and lowest values; thus, a comparison between each bar height and the legend

are displayed instead. A user may use this question to analyze statistical data to compare to their results.

In general, there are significant relations that are established in any given graph. The fourth question is used to indicate which tokens are parts of a graph's relationships. For this question, the user inputs keyword(s), same as the previous questions and the relevant results are presented, including some tokens related to the relation between the X- and Y-labels (see examples in Fig. 9). Then, the users must interpret the graph relations and expressions.

The graphs used in this system were collected from a number of publications, and they are always described by captions and cited paragraphs. Sentences comprise several tokens that are dependent on one another. The fifth question is similar to Question 4, but the question investigates the relationships between two different keywords. A user may use this question to understand any implicit relations between two tokens hidden in the descriptions.

The sixth question presents information in bar graphs that feature multiple data labels. The question presents a comparison based on bar heights and legends in the bar graphs. The comparisons can be achieved with respect to one of two items: with respect to bar categories (e.g., X-label) or with respect to the legend (or data label). A user may use this question for data comparison and analysis. Fig. 10 shows an example of results generated using Question 6.

IV. Evaluation

A. System Evaluation Background

This study aims to create the ontology-based graph search engine. Regarding system validation, we used a traditional search engine to compare with our proposed system. We decided to examine other search engine software commonly used nowadays. It should be noted that the classic search engines used for comparison must support full-text search and indexability.

After we investigated several search system software, it seemed to be difficult to pick the best one, if we did not know their functionalities and indexing processes. We determined to select five software containing different functionalities, i.e., ES, Solr, Sphinx, and DB2 text search extender.

Among the classic search engine candidates, ES was the winner that had been used for comparing with our ontology-based graph search engine. ES, which is a full-text search engine, provides retrieved results based on keywords. In experiments, the user independently inputted keywords, selected questions, and some settings to the evaluated systems. Retrieved results from both systems were sometimes different because the users could obtain extended information from our system. For example, users would like to know about a number of publications published in this year compared to last year. They inquire our ontology-based search system by selecting a question asking about tendency and input some relevant keywords, such as paper and number. In this case, it was possible to obtain results from our system. On the other hand, there was a limitation on ES. It could not deal with other specifications besides the inputted keywords. Therefore, this was a huge difference between our system and the traditional one.

B. Experiment Configuration

The model of configuration for the experiments was described in this section.

The database software used for the system was PostgreSQL, which is a sophisticated open-source database management system (DBMS). We used it because of its simplicity and supporting almost all SQL constructs, such as sub-queries. This database used to record the experiment results and user evaluation.

PHP and JavaScript implemented the web application. It connected to the database to fetch graph information for the traditional system and store the results of experiments. In addition, for our search system, the web application must use the ontology that can be launched by Apache Fuseki, which is a SPARQL server providing provides REST-style SPARQL HTTP Update, SPARQL Query, and SPARQL Update using the SPARQL protocol over HTTP.

The users should specify keywords that were in the domains of biology and computer engineering. The data collected from publications were published in those areas; therefore, to obtain available results from the systems, the keywords related to those study areas unavoidably.

These were possible keywords in the biology domain. Most images gathered from publications relating protein, DNA, and diseases. The examples are displayed below.

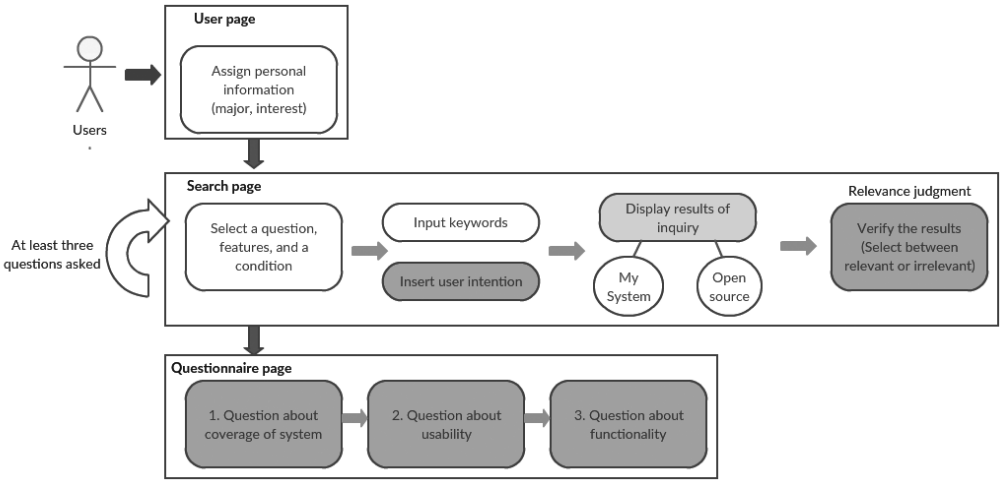

Fig. 11. System flow in feedback mode.

- comparison, miRNA, prediction, protein
- correlation, PDB chain
- amino, alanine
- HbA1c, patients, correlation
- Cancer

Keywords in a computer domain related to data mining and machine learning domains, particularly about algorithms and their performance such as ANNs and SVMs. The examples are displayed below.

- intrusions, classifiers, performance
- image, classification
- sensitivity, specificity
- SVMs, accuracy
- Neural Networks, performance

C. Experiment Procedures

The feedback mode has been described here. In the feedback mode, some additional processes had been required to be inputted by the users. They, who participated the system evaluation, should follow a flow of the feedback mode as shown in Fig. 11.

In the first step, the users should access to a user page to fill own personal information, such as a name, a major, and interest(s). The name represents a display name or username, which does not necessitate being a real name. The users must select a major that they belong. The interests represent topics that they experience with, such as data mining, computer vision, and programming.

Next, the process moved to the search page. The users should test the search engine system on this page. Due to validate the system, we required them to repeatedly test the system totally three times in a row by using different keywords, questions, and other settings because we needed to collect a number of user evaluations for reliable results analysis. Remind that the users should evaluate the obtained results by selecting a decision between relevance and irrelevance. Optionally, we asked the users to input an intention to describe why they used the keywords and what their intention was. This information should be used for the system's discussion. Moreover, on this page, not only our search system was proposed, but the traditional search engine system was also presented here. Therefore, the users must validate the results acquired by both systems. The user responses were collected in the database as well as were used to analyze and compare performance between them.

After the users completed their evaluation in each iteration, the users must submit a button locating at the bottom of the page in order to send the feedbacks to store into the database. Note that, to count as one time queried, the number of results should not be zero; otherwise, the users must select other keywords.

Finally, the feedback process moved to the page, a questionnaire page. The users could give a score or left comments to questions. We used a technique called visual analysis scale (VAS) to scale scores to each question. The submit button locating at the bottom of the page should be clicked after all questions were completely fulfilled.

D. User Feedback Evaluation

In this section, we mainly described the experiment results, evaluation, and analysis.

About the participants, 10 participants attended the evaluation process. According to the limitation of the domains of the dataset, the participants should study or experience about either biology or computer, because they needed to consider the obtained results whether relevance or irrelevance based on their keywords and intention. One participant has an experience about biology, and the rest have studied about the computer. Their nationality is Thai, and their age is between 25-35 years old. An average of ages is 31 years old. A standard deviation is 3.03. Two participants are male, and the rest are female. However, gender is a trivial factor for our evaluation.

Participant	Keywords
1	Protein,antibody
	PBMCs
	cancer
2	sensitivity,specificity
	image,training
	performance,prediction
3	accuracy,comparison
	temperature,comparison
	image
4	emotions
	accuracy,sensitivity,specificity
	heart rate
5	ANN,error
	accuracy,precision
	outlier,ANN
6	predict,regression,coefficient
	predict,back propagation
	linear,regression,predict,ANN
7	image,classification
	comparison,accuracy
	test,training,SVM
8	cluster
	test,training
	comparison,performance
9	intrusions,classifiers
	comparison,SVM,ANN
	error rates,prediction
10	SVM,ANN,comparison
	decision tree,accuracy
	image,classification,accuracy

Fig. 12. Selected keywords for each participant and experiment iteration.

Participant	Elasticsearch			Ontology		
	Precision	Recall	F-measure	Precision	Recall	F-measure
1	0.067	1.000	0.125	1.000	1.000	1.000
	1.000	1.000	1.000	1.000	1.000	1.000
	0.700	0.700	0.700	0.900	0.900	0.900
2	0.500	0.214	0.300	1.000	0.143	0.250
	0.125	1.000	0.222	1.000	1.000	1.000
	0.583	0.389	0.467	1.000	1.000	1.000
3	1.000	0.667	0.800	0.667	0.381	0.485
	0.706	1.000	0.828	0.800	0.571	0.667
	0.000	0.000	0.000	1.000	1.000	1.000
4	1.000	0.500	0.667	1.000	0.500	0.667
	0.143	0.333	0.200	1.000	0.667	0.800
	0.000	0.000	0.000	0.000	0.000	0.000
5	0.813	0.481	0.605	0.920	0.852	0.885
	0.000	0.000	0.000	1.000	1.000	1.000
	0.167	1.000	0.286	1.000	1.000	1.000
6	0.333	1.000	0.500	1.000	1.000	1.000
	0.120	1.000	0.214	1.000	0.667	0.800
	0.389	1.000	0.560	1.000	0.571	0.727
7	0.286	0.400	0.333	0.750	0.600	0.667
	0.900	0.692	0.783	1.000	0.385	0.556
	0.526	1.000	0.690	1.000	0.600	0.750
8	0.800	0.500	0.615	0.667	0.500	0.571
	0.500	0.667	0.571	1.000	1.000	1.000
	0.875	0.875	0.875	1.000	1.000	1.000
9	0.300	1.000	0.462	1.000	0.333	0.500
	0.455	1.000	0.625	1.000	0.200	0.333
	0.667	1.000	0.800	1.000	0.143	0.250
10	0.278	1.000	0.435	1.000	0.200	0.333
	0.455	0.833	0.588	1.000	0.167	0.286
	0.200	1.000	0.353	1.000	0.500	0.667

Fig. 13. Statistical results analyzed by three performance models: precision, recall, and F-measure.

For the results and analysis, as our dataset, the total number of data is 636 images separated to biology 138 images and computer 498 images. There were two types of graphs: 170 images for bar graph and 466 images for

2Dcharts. The images were collected from several publications. Fig. 12 presents the keywords that each participant selected for each experiment iteration. We used this dataset to check our assumption in this study; then, a size of the dataset was not the main point.

After 10 participants completely tested and evaluated on both systems. We statistically analyzed the results by computing three performance measurements: precision, recall, and F-measure, as demonstrated in Fig. 13. Precision is a ratio of retrieved instances identified as relevance. The recall is a ratio of relevant instances that are retrieved. Both performance models are good measurements to deal an imbalanced dataset. For example, the data with relevant class is very rare compared to the irrelevant ones; in another word, the precision and recall depends on how rare is the positive class existed in the dataset, and they are mostly used when the positive class is more interesting than the negative one. Moreover, F-measure is a mean between precision and recall representing an accuracy of the test and how the quality of the system.

Hereafter, we showed the results, including their critical viewpoints.

Based on our observation in Fig. 12 and 13, we noticed that even the participants selected the same keywords; the performance models might not be equal. For example, the keywords from Participant 3 at the first iteration and Participant 7 at the second iteration were defined as "accuracy, comparison". This situation was happened because of two reasons. First, particular settings had been performed. Illustrating that a participant might choose a condition that allowed only results typed as a bar graph; whilst another participant did not set any condition. Then, the obtained results might differ due to the different settings. Second, the participants were determiners to decide the results whether relevance or irrelevance; thus, the decision might be different depending on their consideration.

Before we proceeded the experiments, we defined a hypothesis that such results retrieved from the ontology-based search system should be outperformed than the traditional one, i.e., Elasticsearch (ES). Most results (Fig. 13) agreed our hypothesis, but a few results did not. The ontology-based search engine could acquire the relevant results by using AND operator; since they certainly matched to an intention of participants. Unfortunately, a number of retrieved results were sometimes too small because only exact matches had been obtained by the systems that caused a small amount of recall. As the recall from Participant 9 and 10, the participants selected some specific keywords, and only one result was acquired on each iteration. They decided it as relevance; hence, the precision was high, as opposed to recall. In our dataset, there were some results relevant to the keywords, but they could not show on a screen. They could not find certain keywords, but their synonyms or related words had been discovered. For example, Participant 10 selected the “decision tree, accuracy” keywords. She needed to examine the accuracy of the decision tree algorithm. Note that several documents collected in the dataset indirectly mentioned about decision tree algorithm. They used the decision tree algorithm name instead, e.g., J48. Our system could return the result containing both keywords but could not for J48. ES could obtain a number of results that related to "accuracy" which accidentally matched to J48. Therefore, the recall of ES was surely higher, but the precision was lower than our system.

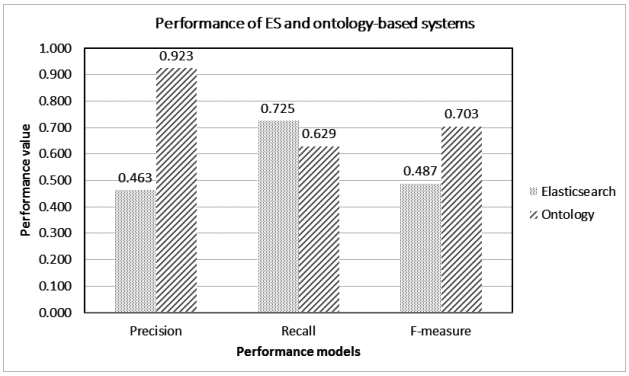

Fig. 14. Average precision, recall, and F-measure from ES-based and ontology-based search engine systems.

The precision and recall from Participant 4 in the last iteration were zero because no relevant results were returned from both systems. She used a keyword "heart rate"; unfortunately, there was not any data in our dataset relating to the "heart rate" Based on our inspection, most returned results were regarded as heart disease and did not mention about heart rate. Moreover, this issue also happened with Participant 3 in the last iteration of the ES-based system. She selected a very simple keyword "image" to find graphs that related about image. However, the number of results retrieved from ES was zero because of her query setting. She required only the image that the "image" keyword existed inside the graph, for instance, in X- or Y-titles. The ES-based system could not support this requirement. This was an evidence that our system could handle this specification, and the participants could literately obtain the relevant results.

Regarding the performance of both systems, we briefly analyzed and computed the results with the three performance models as average values, as presented in Fig. 14. Obviously, the precision of our system was much higher than the ES because the participants considered that our system could mostly provide relevant results by using the specific questions, condition, and features; meanwhile, the ES-based search engine system provided the results based on only given keywords. However, the recall of our system was lower compared to the ES-based system, because currently, the ontology-based system did not support synonym or related words. Fortunately, this problem could be solved simply by connecting to other ontologies, such as DBpedia, to inquire about other related words and use them as extra keywords.

To compare the performance between both systems, this was difficult to use either precision or recall to consider the system performance. Therefore, we computed the F-measure, which is the harmonic mean of precision and recall. After we analyzed it, the F-measure from our system was clearly much higher than the ES one. In general, the high F-measure represents the better system performance. In addition, on the questionnaire page, the participants gave scores to questions asking about system coverage, usability, and functionality. An average score of Question 9 was one of supportive evidence to evaluate the system performance (Fig. 16). Hence, our system was outperformed than the ES-based

system confidently.

Fig. 15 shows a list of questions. Question 1, 2, and 3 represent system coverage. For example, "do the provided questions cover your need for inquiry the system?" Question 4 to 11 ask about system usability, such as how suitable a layout of the user interface, how speed, how accuracy, and how useful to a study. The rest, i.e., Question 13, asks about functionality, such as error handling. Note that some question numbers are skipped because they are comments.

We considered the obtained scores of each question. We focused on Question 1, 2, 3, 9, and 11 because they were very important questions to validate the system. We assumed a range of satisfaction showing as follows:

- 100-80 = Very satisfied
- 79-60 = Satisfied
- 59-40 = Neural
- 39-20 = Poor
- 19-0 = Bad

The average scores from those questions were classified as Very satisfied. This could be concluded that our system was suitable to open up a new way for a novel technique of information retrieval. Not only the high performance was presented as described above, but the participants also felt comfortable to use the system because it could support the research studies of the participants as displayed in Fig. 16 at Question 11.

Question numbers	Questions
q1	Do the provided questions cover your need for inquiry the system?
q2	Do the provided conditions cover your need for inquiry the system?
q3	Do the provided features cover your need for inquiry the system?
q4	Is the system easy to use?
q6	Ask about system presentation, is the system suitable to use for information inquiry and its results representation?
q8	How do you think about the successive seeking time of the system?
q9	Do you think that the system provides accurate results from your queries?
q10	To prove the system validation, do the system provide wider information due to using ontology when comparing the traditional search engine?
q11	Is the system applicable or useful for your study?
q13	Is the system able to handle errors, unexpected situations, or capture any anomalies?

Fig. 15. List of questions in the questionnaire page.

Here, we analyzed scores representing the satisfaction values provided by the participants. Fig. 18 depicts the assigned scores of questions for each participant. As our observation, a participant gave some comments because she thought that the system should improve somehow due to less score of Question 1, 2, and 11 obtained by Participant 3. Her scores were not in a normal range of standard deviation (Fig. 17). Her opinions were about a small volume of the dataset. As described our data collection, the size of the dataset was around 600 data; since she possibly did not obtain any result from the system if she used too specific keywords. Moreover, she is interested in video comparison and temporal comparison; unfortunately, our computer dataset domain was only about data mining and machine learning.

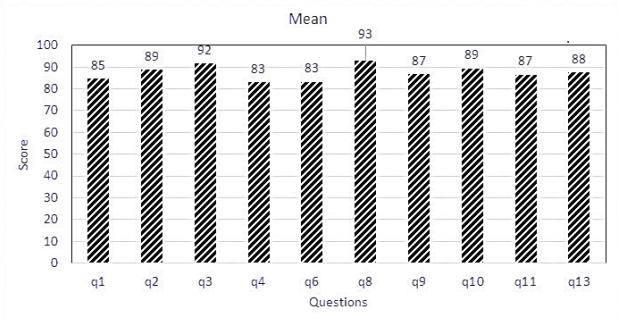

Fig. 16. Mean of scores provided by participants in the questionnaire page.

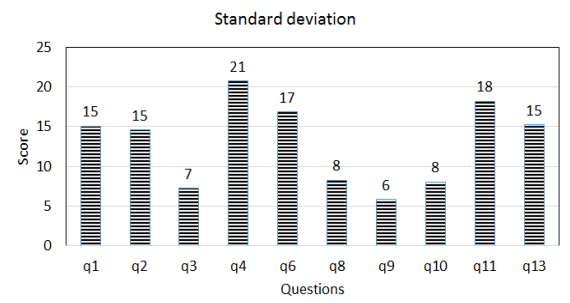

Fig. 17. The standard deviation of scores provided by participants in the questionnaire page.

Participant	Score									
	q1	q2	q3	q4	q6	q8	q9	q10	q11	q13
1	91	94	94	90	80	91	85	90	95	100
2	90	95	95	95	90	100	95	91	95	89
3	50	58	80	50	50	100	75	80	40	50
4	100	100	100	100	100	89	90	100	100	100
5	90	100	90	90	100	100	85	85	90	100
6	91	99	99	100	81	100	91	100	99	82
7	85	81	83	90	87	80	86	87	77	85
8	80	70	85	40	60	90	80	75	80	80
9	70	90	90	85	100	80	90	90	100	95
10	100	100	100	91	85	100	90	95	90	95

Fig. 18. Scores of each question in Questionnaire page provided by 10 participants.

Then, the precision of our system slightly rose from 0.923 to 0.935. The recall of ES-based system reduced after the outlier was omitted that caused the similar recall value to the ontology-based search engine system. However, F-measure of both systems trivially decreased, but the difference of the value was not changed. Fig. 19 depicts the true performance of both search engine systems that already omitted outliers from the results.

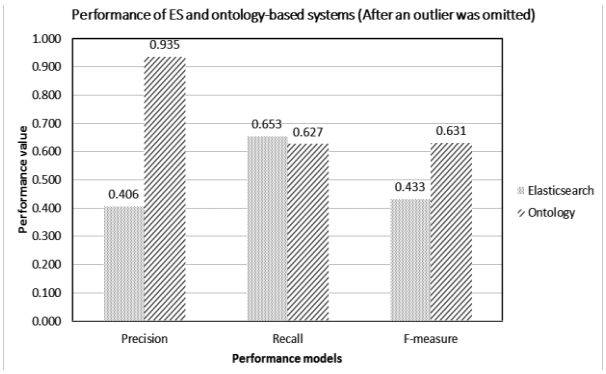

Fig. 19. True performance of the search engine system without outliers.

V. Discussion

The critical point needed to be discussed here was how the system can achieve the problem of the semantic gap. The main idea of this research was to propose methods extracting information from graphical and linguistic representations as well as utilize them to express explicit and implicit knowledge. We used the method of graph

component extraction [5] and graph information extraction [6] to extract information from the graphical data, including graphs' linguistic data, e.g. caption. Then, they were recorded into ontology; since the system was an application to show the results from the ontology.

As regards the ES-based system, it was a keyword-based search system; since, it could not provide the information located in graphical images, only image descriptions available.

We measured the true performance of the whole systems corresponding to the results obtained from the system that already omitted outliers. As the results, the precision was greatly high comparing to the ES-based system, 0.935 and 0.406 respectively. However, the recall of our system was slightly lower. To analyze the recall of both systems, our system offered a lower value, but it was almost similar to the recall of the ES-based system. In another word, the difference between those recall values, which was equal to 0.02, was very trivial. Thus, if we analyzed the recall only, the performances from both systems were very similar. Regards the F-measure, our system provided higher F-measure than the other did. This means that our system represented the better performance. To sum up, based on the evidence described here, our ontology-based search engine system can overcome the addressed problem, because it offers higher precision and F-measure comparing to the ES-based search engine system.

Regarding the target data of this study, a collection of graph images had been used for the experiments of each system. The data were gathered from the scientific literature. This system had covered computer science and biology domains. The graphs from different domains provide a diversity of data expression. In a viewpoint of the data domain, a possible solution to handle the variant data expression is to integrate ontologies from other's domains, such as Physics or Biology. In contrast, about different graph structure aspect, our system can extract precise information from graphs with the two-dimensional axes graph, not including tree graph or network. To deal with other kinds of graph structures, it is necessary to propose particular methods to extract information from them because of a diversity of graph expression existed. There are many ontologies publishing on the Internet. To extend our ontology, we need to merge ours and other ontologies together. A coverage of the system should depend on what domain of the ontologies is integrated with. However, to integrate them, this is necessary to take into account to ontology alignment. Kinds of interoperability are limited because a minimal change has been required for ontology schemes in order to merge inter-ontologies. Thus, it is important to standardize our ontology scheme compatible with the merged ontologies. To do so, before creating the ontology, we should examine the schemes of merged ontologies in advance and attempt to seek what concept can be connected. Moreover, a merging process can be performed in many particular ways, such as manually, semi-automatically, or automatically. Manual ontology merging is highly labor-intensive; hence, semi or entirely automated techniques are definitely preferable. To do this, a similarity of concept relationships should be examined. A merging system traces along relationships through ontologies and observes which parts contain similar concepts and relationships. In addition, they may realize the similarity of concepts through textual string metrics, e.g., edit distance, including semantic knowledge and relationships. There are many kinds of graphs available in the literature. In this study, we limited to a kind of graph presented as a general structure, such as bar graph and plot graph, because they have been often used in the scientific literature rather than other graph types. They are suitable to convey the statistical data or compare results. Only two types of graphs have been used in this research: bar graph and 2Dchart. The system is highly applicable to these data supportive by obtaining high accuracy as shown in the experiments. However, if we deal with either bar graph or 2Dchart, system performance may increase somehow because of no classification errors.

This system utilizes entire systems proposed in our previous studies. Regarding limitations, this system does not support a lemma technique yet. Note that the lemma is a technique to change a word to its root. There are some libraries available on the Internet. If we integrate a lemma process to our system, we suppose that the obtained results should be enlarged, and new knowledge is delivered. Moreover, it cannot separate between stop words or rare words. This problem will be solved if we use text-mining technique. The size of the dataset was limited and specified only two domains. To cover the users' needs, the data volume should be expanded. Our ontology should be integrated with other ontologies to enlarge data source. In Question 2 of this system, we investigated the main idea based on sentences containing keywords and the first sentence of the paragraph. However, to obtain the main idea precisely, we should utilize text summarization, which is a text mining technique, to summarize the whole paragraphs and show only a core part of paragraphs.

To enhance the system usability, a keyword or spelling suggestions may be necessary for the users, who do not know how to spell the keywords or do not have ideas to select keywords. Due to this, using data mining may be a good solution, because it analyzes users' behaviors and suggests possible keywords. Additionally, we obtained the unexpected findings by observing the results of relationships. The partial relationships should be useful if we input them into ontology because we suppose that we may discover new knowledge by tracking other relations on the ontology. In another aspect, we may cluster the graph relationships based on their shared relationships by using a graph or network clustering and find some similarities on the graphs belonging to the same group. Moreover, if we utilize deep learning to the system, it is possible to develop a question answering system based on our ontology. This function surely facilitates users to obtain desired answers speedily. Further, the deep learning is used for matching between text and image. They represent as vectors and using a deep learning technique (e.g., Convolutional Neuron Networks) analyzes and matches the two vectors. If this is used to our system, the obtained results will be unlimited to only graphs but included other kinds of images. For example, a user needs to query the system by using a keyword "compiler", our system will provide graphs showing statistical data about "compiler", including other images, such as compiler pictures. Currently, this system did not have a ranking

feature.

To order relevant graphs or documents, we will use deep learning to rank the results by analyzing user interactions, such as a click. Furthermore, based on the system's ability, it is possible to develop a new function integrated into our system to suggest or recommend publications to readers. When they use our existed system to query relevant graphs corresponding to their keywords, some relationships have been discovered in the graphs. The new function recommends the publications corresponding to the relationships. Since the readers can decide which documents are worth to read. Regarding the ontology creation, the ontology scheme may be able to deduce by data itself. If a system can analyze the data and result in some existed concepts and relations, it is possible to create the ontology scheme automatically.

During the experiments and evaluations, we received many useful feedbacks and comments from the participants in order to improve the system usability as follows:

- At the result section of the search page, we should include sources of documents, such as a publication URL and a paper's title.
- We should enlarge the size of the dataset, including expanding data domains to cover all needs.
- We should redesign option selections in the search page to be simpler.
- The layout of the prototype should be organized to prevent confusion.

As the comments above, they required an interface improvement to support user convenient. The participants did not deny the idea of method and my assumption supported by results from questionnaire and evaluation.

VI. Conclusions

We found that significant problems had been solved by the system created in this study. The technical contribution was to propose a technique of ontology usage by using both graphical and textual information with the search engine system. The ontology presented relationships among these data. We principally addressed the problem of the semantic gap. We clarify that this idea of implementation helped to reduce a semantic gap between that information. We conducted several experiments and evaluated the obtained results. It clearly showed that the system could identify and extract information from the graph image. Moreover, the information was included in ontology integrated into the search system. As the results, our system can provide the information to users via the ontology. Since we clarified that the problem of the semantic gap was already solved by this research. We programmed a web-based application applicable to search and query thought our constructed ontology created by all extractable graph information.

Ten participants helped us to evaluate the systems by selecting specific questions, settings, and input some keywords. They decided the returned results by either relevance or irrelevance. We validated the performance between our ontology-based and ES-based search engine systems. As the results, we concluded that our ontology-based search engine systems provided better performance than the traditional one due to higher F-measure obtained. Moreover, the result from a questionnaire was supported in our conclusion. Regarding the limitations of the study, this system has covered the data from computer science and biology domains. However, it is also applicable to other domains if we expanded the target data. Due to graph types and a kind of graph limited, it can express the information extracted only from bar graph and 2Dchart which are in a general graph structure. In conclusion, we proposed the systems to extract the graphical and linguistic information from the graph image itself and its descriptions. The system provided the great performance measurements; since it proved that it could mitigate the semantic gap problem and achieve entire objectives. It clarified that the ontology-based search engine system provides precise and concise graph information outperforming than traditional search engine systems. The major contribution is not only the new method of ontology-based search engine system but also an ontology design supporting graph information and descriptions.

For the future study, the ontology-based search engine system should integrate a keyword suggestion to recommend possible keywords to users. It will utilize an intelligent technique, e.g., deep learning, to analyze user behaviors and suggest them the keywords. Another idea is to analyze description context to predict the user intention and offer some possible keywords. Further, a question answering system will be introduced by deep learning in the future. If these functions will be proposed, the ontology-based search engine is surely much more powerful. Moreover, in the future, A system will generate the ontology automatically by referring to some existing structures and relationships, such as dependency parsing in sentences.

References

[1] H. Bast, A. Chitea, F. Suchanek, and I. Weber, "Ester: Efficient search on text, entities, and relations," in *Proc. the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2007, pp. 671–678.

[2] Y. Tsuruoka, J. Tsujii, and S. Ananiadou, "FACTA: A text search engine for finding associated biomedical concepts," *Bioinformatics*, vol. 24, no. 21, pp. 2559–2560, 2008.

[3] M. Batko *et al.*, "Building a web-scale image similarity search system," *Multimedia Tools and Applications*, vol. 47, no. 3, pp. 599–629, 2010.

[4] J. Ma, "Content-based image retrieval with hsv color space and texture features," in *Proc. International Conference on Web Information Systems and Mining*, 2009, pp. 61–63.

[5] S. Kanjanawattana and M. Kimura, "Extraction and identification of bar graph components by automatic Epsilon estimation," *International Journal of Computer Theory and Engineering*, vol. 9, no. 4, 2017.

[6] S. Kanjanawattana and M. Kimura, "Extraction of graph information based on image contents and the use of ontology," in *Proc. International Conferences ITS*, 2016, pp. 19–26.

[7] S. Kanjanawattana and M. Kimura, "A proposal for a method of graph ontology by automatically extracting relationships between captions and X- and Y-axis titles," in *Proc. the 7th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management*, 2015, vol. 2.

[8] S. Carbon *et al.*, "AmiGO: Online access to ontology and annotation data," *Bioinformatics*, vol. 25, no. 2, pp. 288–289, 2008.

[9] M. Gao, C. Liu, and F. Chen, "An ontology search engine based on semantic analysis," in *Proc. Third International Conference on Information Technology and Applications*, 2005, vol. 1, pp. 256–259.

[10] D. Bonino, F. Corno, L. Farinetti, and A. Bosca, "Ontology driven semantic search," *WSEAS Transaction on Information Science and Application*, vol. 1, no. 6, pp. 1597–1605, 2004.

[11] G. Li, S. Ji, C. Li, and J. Feng, "Efficient fuzzy full-text type-ahead search," *The International Journal on Very Large Data Bases*, vol. 20, no. 4, pp. 617–640, 2011.
[12] T. Jayalakshmi and C. Chethana, "A semantic search engine for indexing and retrieval of relevant text documents," *Int. J*, vol. 4, no. 5, pp. 1–5, 2016.
[13] A. Both, A.-C. N. Ngomo, R. Usbeck, D. Lukovnikov, C. Lemke, and M. Speicher, "A service-oriented search framework for full text, geospatial and semantic search," in *Proc. the 10th International Conference on Semantic Systems*, 2014, pp. 65–72.
[14] J. Brophy and D. Bawden, "Is Google enough? Comparison of an internet search engine with academic library resources," *Aslib Proceeding*, vol. 57, no. 6, pp. 498–512 *s*, 2005.
[15] M. A. Hearst *et al.*, "BioText Search Engine: Beyond abstract search," *Bioinformatics*, vol. 23, no. 16, pp. 2196–2197, 2007.
[16] S. Arivazhagan *et al.*, "Ontology-based personalized search and browsing," *Pattern Recognition Letters*, vol. 25, no. 2, pp. 156–161, Feb. 2003.
[17] H.-M. Müller, E. E. Kenny, and P. W. Sternberg, "Textpresso: An ontology-based information retrieval and extraction system for biological literature," *PLoS Biol*, vol. 2, no. 11, p. e309, 2004.
[18] H. Bast, B. Buchhold, E. Haussmann, and others, "Semantic search on text and knowledge bases," *Foundations and Trends® in Information Retrieval*, vol. 10, no. 2–3, pp. 119–271, 2016.
[19] E. Hyvönen, S. Saarela, and K. Viljanen, "Ontogator: Combining view-and ontology-based search with semantic browsing," *Information Retrieval*, vol. 16, p. 17, 2003.
[20] H. Zhong and L.-M. Xia, "Ontology-based image retrieval," *Computer Engineering and Applications*, vol. 42, no. 17, pp. 37–40, 2007.
[21] T. M. Deserno, S. Antani, and R. Long, "Ontology of gaps in content-based image retrieval," *Journal of digital imaging*, vol. 22, no. 2, pp. 202–215, 2009.
[22] V. Mezaris, I. Kompatsiaris, and M. G. Strintzis, "An ontology approach to object-based image retrieval," in *Proc. International Conference on Image Processing*, 2003, vol. 2, p. 511.
[23] R. Zhao and W. I. Grosky, "Narrowing the semantic gap-improved text-based web document retrieval using visual features," *IEEE Transactions on Multimedia*, vol. 4, no. 2, pp. 189–200, 2002.
[24] Y. Chen, H. Sampathkumar, B. Luo, and X. Chen, "iLike: Bridging the semantic gap in vertical image search by integrating text and visual features," *IEEE Transactions on Knowledge and Data Engineering*, vol. 25, no. 10, pp. 2257–2270, 2013.
[25] S. Kanjanawattana and M. Kimura, "Novel ontologies-based optical character recognition-error correction cooperating with graph component extraction," *Broad Research in Artificial Intelligence and Neuroscience*, vol. 7, no. 4, pp. 69–83, 2017.

Sarunya Kanjanawattana was born in Naknonratchasima, Thailand, in 1986. She received the B.E. degree in computer engineering from Suranaree University of Technology, Nakhonratchasima, Thailand, in 2008, and M. Eng from Asian Institute of Technology, Pathum Thani, Thailand in 2011. In 2017, She graduated from her doctor course with a major of functional control systems from Shibaura Institute of Technology, Tokyo, Japan. In 2011, she joined National Electronics and Computer Technology Center, Thailand, as a research assistance. Her project related to finding an optimal solution to traffic congestion. At present, she works at Suranaree University of Technology as a lecturer in the Department of Computer Engineering. Her research interests included data mining, machine learning, natural language processing, ontology, and computer vision.

Masaomi Kimura received a B.E. degree in precision engineering in 1994, and M.S. in 1996, and D.S. degrees in 1999 in physics from University of Tokyo. He is a professor in the Department of Information Science and Engineering, Shibaura Institute of Technology. His current interests include data engineering, complex systems, and medical safety.