IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received November 9, 2018, accepted December 28, 2018, date of publication January 10, 2019, date of current version January 29, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2891710

Intelligent Health Diagnosis Technique Exploiting Automatic Ontology Generation and Web-Based Personal Health Record Services

GUN-WOO KIM AND DONG-HO LEE
Department of Computer Science and Engineering, Hanyang University, Seoul 04763, South Korea

Corresponding author: Dong-Ho Lee (dhlee72@hanyang.ac.kr)

This work was supported by the National Research Foundation of Korea funded by the Ministry of Education, Science, and Technology through the Basic Science Research Program under Grant NRF-2016R1D1A1A09918271.

ABSTRACT Growing interest in healthcare has promoted the use of symptom checkers, which are online health applications that provide diagnostic information on users' health. However, their diagnostic accuracy remains low because the existing symptom checkers rely on manually constructed knowledge models through labor-intensive processes or perform diagnoses based on simple pairwise relationships between diseases and symptoms without considering personal health conditions. In this paper, we propose an intelligent health diagnosis technique that exploits automatically generated ontology and Web-based personal health record services. The proposed technique first automatically generates a human disease diagnosis ontology by exploiting two well-established ontologies for diseases and symptoms: a large-scale medical bibliographic database and an open biomedical repository. When a user enters the symptom-based queries, possible diagnoses are identified by analyzing the user's queries and their health record data via semantic inferences of the automatically generated ontology. Subsequently, the ranked diagnostic results are provided to the user via ranking methods that consider the user's symptoms, personal health attributes, and multi-level diagnosis. The proposed technique also provides the user's diagnostic progress information, which can be used to track or monitor the progress of diseases by considering changes in symptoms over time. The proposed technique was evaluated through a comparison with the existing well-known symptom checkers and other related approaches. The evaluation results show that the proposed technique can feasibly help to improve diagnostic accuracy and deliver appropriate diagnostic information for healthcare action by users.

INDEX TERMS Health information retrieval, healthcare, human disease diagnosis ontology, intelligent health diagnosis, personal health record.

I. INTRODUCTION

In today's e-health era, the internet has become a common means of acquiring knowledge on personal health. However, most healthcare consumers primarily use search engines to retrieve health information based on their symptoms without precise consideration of their personal health conditions [1], [2]. Thus, they can easily confuse or misunderstand their health status by reviewing only high-ranked search results without filtering out irrelevant or unreliable health information. This can result in social issues such as *cyberchondria*, which is the unfounded escalation of concerns about common symptoms based on search results [3].

To acquire more reliable information, there has recently been a proliferation of online health applications called symptom checkers. Symptom checkers are sophisticated health applications that attempt to more effectively provide potential diagnostic information regarding a user's symptoms. Thus, they can help users who lack proper medical knowledge to more easily understand their individual health concerns (i.e., what the symptoms could mean) and direct them to the appropriate care settings or inform them as to whether they should seek care at all [4].

However, despite the benefits and proliferation of symptom checkers, concerns have been raised in many recent studies regarding their diagnostic accuracy. These studies expressed skepticism regarding the usefulness of diagnostic results suggested by symptom checkers due to their low diagnostic accuracy [5]–[7]. For example, one study evaluated the accuracy of 23 existing symptom checkers and found that they provided correct diagnoses in only 34% of test cases [7]. Diagnostic accuracy is a significant issue that affects user satisfaction and confidence in the validity of symptom checkers. Therefore, enhancing the value of these applications will require the development of more accurate and reliable diagnosis techniques. The primary reasons for low diagnostic accuracy in current symptom checkers can be summarized as follows.

A. MANUAL CONSTRUCTION OF KNOWLEDGE MODELS

In general, symptom checkers require knowledge models that encoded the relationships between diseases and symptoms to obtain diagnostic results. Such models constitute a dominant proportion of symptom checkers and are used to examine diseases that are related to user-entered symptoms [8]. However, most knowledge models are manually constructed, which requires tremendous amounts of time and effort from medical experts [9]. Hence, this makes it costly to construct usable knowledge models, and the knowledge could be biased or brittle. Such models are also difficult to adapt or update for new diseases or symptoms.

B. LIMITED TERMINOLOGY PROCESSES

Typically, users enter a series of symptoms to obtain diagnostic information from a symptom checker. However, due to the peculiarities of medical terminology, user-entered symptoms can be represented using a variety of synonyms (e.g., *headache* and *cephalgia*) or abbreviations (e.g., *CP* for *chest pain*). Users might even use codes (e.g., *ICD-10-CM R05* for *cough*) extracted from medical prescriptions to represent their symptoms. Therefore, symptom checkers must be able to handle a broad range of lexical variations for each user-entered symptom to provide proper diagnostic information. However, most symptom checkers cannot recognize a user-entered symptom that does not precisely match a symptom term encoded in their internal knowledge models. Specifically, they are often unable to recognize abbreviations and codes. Thus, current symptom checkers manifest low diagnostic accuracy because they provide results by ignoring unrecognized user-entered symptoms.

C. DISREGARD OF PERSONAL HEALTH

Symptom checkers should reflect the user's personal health in the diagnostic process because each user has different characteristics and health conditions. However, a study conducted by Semigran *et al.* [7] found no accuracy difference between symptom checkers that asked for and those did not ask for personal health. This is because most existing symptom checkers either disregard or do not effectively incorporate personal health attributes such as user demographic information and well-known measures of health (e.g., blood glucose and blood pressure levels). Consequently, these symptom checkers only provide diagnostic results on common diseases that have simple pairwise relationships with user-entered symptoms without including an individual's health. Furthermore, with the increased prevalence of chronic diseases, there is increasing necessity for a multi-level diagnosis approach, which assumes that user's current diseases might be a diagnostic element for another disease [10]. Multi-level diagnosis can be used as an important element to increase the diagnostic accuracy of symptom checkers where diagnosis results are solely based on user-entered symptoms. However, although a multi-level diagnosis is a well-known approach to medical experts as a way to diagnose patients, most symptom checkers have not incorporated this approach until recently.

To address these issues, we propose a novel intelligent health diagnosis technique. The proposed technique aims to provide reliable diagnostic results to users for helping to take appropriate healthcare actions by enhancing the diagnostic accuracy. To this end, we automatically generate an ontology called the Human Disease Diagnosis Ontology (HDDO) and exploit it as a knowledge model. Further, to reflect personal health status in the diagnostic process, we leverage the individual health record data provided by the PHR service, which is a web-based set of tools that enables the active management of personal health attributes (e.g., demographic information, weight, blood glucose, and blood pressure) and medical records (e.g., records of chronic diseases). The proposed technique consists of two components: 1) automatic ontology generation and 2) health diagnosis.

1) AUTOMATIC ONTOLOGY GENERATION

This component comprises methods for the automatic generation of our knowledge model, the HDDO. To automatically generate the HDDO, we exploit three types of biomedical resources: 1) source ontologies, Disease Ontology (DO)[1] and Symptom Ontology (SYMP),[2] 2) a large-scale medical bibliographic database, PubMed,[3] and 3) an open biomedical repository, BioPortal.[4] Although DO and SYMP are well-established ontologies for human diseases and symptoms, several issues had to be addressed prior to exploiting them as our knowledge model. First, they simply classified diseases and symptoms according to their types without considering lexical variations of the terms. Also, since the two ontologies were developed separately for diseases and symptoms, no relationships between diseases and symptoms were established. The relationships between diseases and personal health attributes are significant factors that cannot be overlooked; however, these relationships cannot be derived from only the two ontologies. Therefore, we exploit DO and SYMP only as basic vocabulary sources to define diseases and symptoms terms in the HDDO. Further, to support all lexical variations, we expand the disease and symptom terms by exploiting the terminological knowledge from BioPortal. In order to specify all the relationships necessary for our diagnosis technique, we exploit bibliographic records from

[1] http://disease-ontology.org/
[2] http://symptomontologywiki.igs.umaryland.edu/
[3] https://www.ncbi.nlm.nih.gov/pubmed/
[4] https://bioportal.bioontology.org/

PubMed. Here, we do not use a full-text search of the journal articles or abstracts, but only exploit article IDs and Medical Subject Heading (MeSH) metadata to acquire more precise relationships.

2) HEALTH DIAGNOSIS

This component involves the provision of personalized diagnostic results based on the HDDO and PHR data. First, we analyze the user's symptom-based queries and PHR data and identify possible diagnoses by filtering out irrelevant diseases via semantic inferences of the HDDO. Subsequently, we deliver ranked results of possible diagnoses to the user based on our proposed ranking methods that consider the user's symptoms, personal health attributes, and multi-level diagnosis. This component can elevate the ranking of diagnoses related to user health according to individual health conditions and medical records. Therefore, we can provide personalized diagnostic results to the user, rather than general results. In addition, the symptoms used in the diagnosis can change over time, such as when certain symptoms appear or disappear. Thus, tracking or monitoring the progress of diagnosed diseases or other possible diseases is an important aspect of diagnosis techniques. By storing the user's diagnostic result logs in the HDDO, we provide diagnostic progress information that considers newly added or removed symptoms from previous diagnostic results.

The remainder of this paper is organized as follows. Section II discusses related work. Section III gives an overview of the proposed technique. Section IV presents the methods used for automatic ontology generation and health diagnosis. Section V describes a prototype implementation of the proposed technique. Section VI discusses and analyzes our results and findings. Finally, Section VII presents concluding remarks.

II. RELATED WORK

To enhance the accuracy of health diagnosis techniques, a number of approaches have been introduced, including fuzzy sets, genetic algorithms, neural networks, machine learning, and recommender systems [11]–[14]. Further, a large number of symptom checkers that have applied health diagnosis techniques are currently available, including *AskMD*, *Isabel* and *iTriage*. However, an important issue in health diagnosis techniques is that diagnostic knowledge, such as the relationships between diseases and symptoms or between diseases and personal health attributes, is often vague and difficult to obtain [15]. Therefore, to construct knowledge models, most studies have used manually compiled diagnostic knowledge or patient cases obtained through the participation of medical experts [16]–[19]. However, these approaches are costly and time-consuming tasks that require considerable effort by highly paid medical experts. In addition, to adopt the latest knowledge on diseases and symptoms, additional medical expert participation is required. Another limitation of these approaches is that they have attempted to test only a few specific diagnoses using a limited number of diseases and symptom knowledge due to the difficulty of acquiring reliable large-scale knowledge.

To address these issues, automatic knowledge model construction through diagnostic knowledge extraction from medical documents has gained attention, and it has been currently undertaken in several studies [20]–[22]. Such studies have primarily focused on finding disease and symptom relationships and their correlation weights by exploiting term co-occurrence analysis of text mining.

Mohammed *et al.* [20] presented a linking algorithm to find the relationships between diseases and symptoms using medical documents from online health websites. However, they simply assumed that a relationship exists if the disease and symptom terms appear together in a single document without consideration of lexical variations such as synonyms, and abbreviations. Therefore, their model has low precision and recall performance on diagnostic results because it can extract only a fraction of disease and symptom relations as diagnostic knowledge.

To overcome these drawbacks, Okumura and Tateisi [21] utilized MetaMap, which maps phrases in medical documents to standardized Unified Medical Language System (UMLS) concepts. MetaMap is a NLP tool that can map lexical variations of disease or symptom terms to the same UMLS concept. For example, the word "*hypertension*" and the synonym "*high blood pressure*" can be mapped to the same UMLS concept "*hypertensive disease*". Their methods can overcome the issues arising from the lexical variations of disease and symptom terms, allowing for the coverage of broader relationships. However, MetaMap often fails to capture abbreviations or codes and often assigns unexpected UMLS concepts [23]. Furthermore, MetaMap does not handle negative mentions in medical documents. Thus, extracted disease and symptom relations might be negated in medical documents [24]. As they ignore the extra processes of handling negations such as relations that are stated as nonexistent, they may extract atypical or even unrelated relations as diagnostic knowledge.

Another interesting approach was undertaken by Zhou *et al.* [22]. They assumed that the current knowledge model construction methods based on text analysis cannot yield highly accurate results. Therefore, to construct a knowledge model, they only use journal article IDs and associated MeSH metadata from PubMed without using the full-text search of the medical documents. In PubMed, each article has an article ID and associated MeSH metadata. In particular, MeSH metadata includes a list of keywords describing core topics addressed in the article. These keywords are manually curated by trained experts following standardized procedures, thereby ensuring highly accurate assignments. Therefore, to extract the relationships between diseases and symptoms, they exploit MeSH metadata for disease and symptom terms and check for co-occurrence of the terms in the same PubMed's article ID. This method produced promising results in terms of diagnostic knowledge extraction and is currently used in various approaches in biomedical research, such as

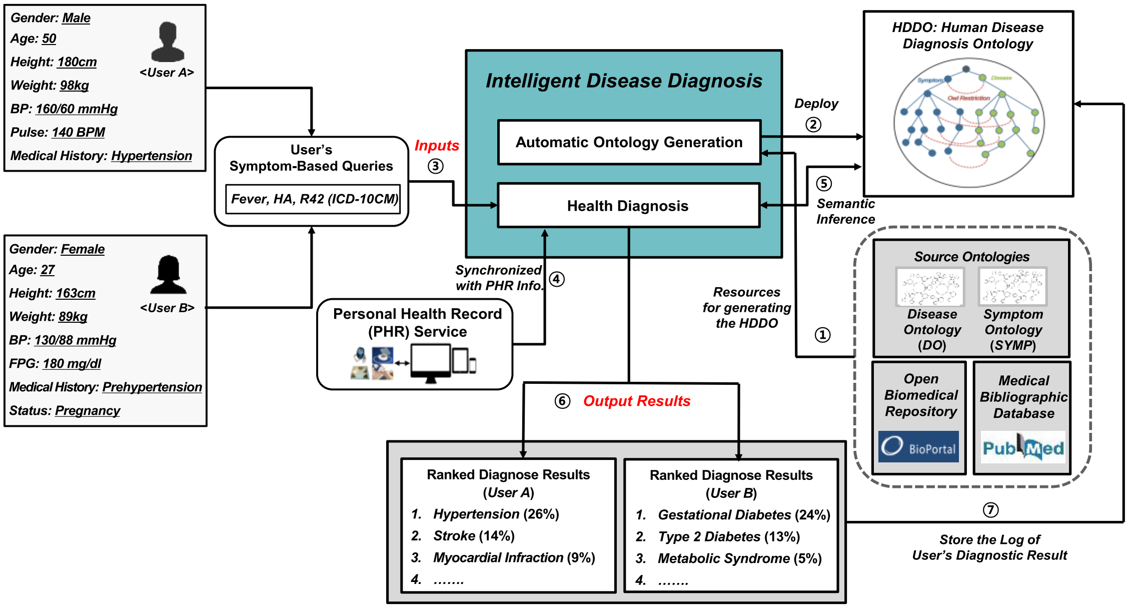

Gender: Male
Age: 50
Height: 180cm
Weight: 98kg
BP: 160/60 mmHg
Pulse: 140 BPM
Medical History: Hypertension
<User A>
Gender: Female
Age: 27
Height: 163cm
Weight: 89kg
BP: 130/88 mmHg
FPG: 180 mg/dl
Medical History: Prehypertension
Status: Pregnancy
<User B>
User's Symptom-Based Queries
Fever, HA, R42 (ICD-10CM)
Inputs ③
Intelligent Disease Diagnosis
Automatic Ontology Generation
Health Diagnosis
Deploy ②
HDDO: Human Disease Diagnosis Ontology
Symptom
Disease
Owl:Restriction
⑤ Semantic Inference
Synchronized with PHR Info. ④
Personal Health Record (PHR) Service
Resources for generating the HDDO
①
Source Ontologies
Disease Ontology (DO)
Symptom Ontology (SYMP)
Open Biomedical Repository
BioPortal
Medical Bibliographic Database
PubMed
⑥ Output Results
Ranked Diagnose Results (User A)
1. Hypertension (26%)
2. Stroke (14%)
3. Myocardial Infraction (9%)
4. ....
Ranked Diagnose Results (User B)
1. Gestational Diabetes (24%)
2. Type 2 Diabetes (13%)
3. Metabolic Syndrome (5%)
4. ....
⑦
Store the Log of User's Diagnostic Result

FIGURE 1. Overview of health diagnosis technique with a brief scenario involving two users.

symptom network [25] and medication design [26]. However, their method only considers the relationships between diseases and symptoms, and the consideration of the relationship between diseases and personal health attributes are left as a challenge to be addressed in their future work. Furthermore, although there are various text mining measures for term co-occurrence analysis, they only showed the experimental results of Term Frequency-Inverse Document Frequency (TF-IDF) to rationalize the accuracy of their method.

To consider personal health attributes as diagnostic knowledge, Rodriguez-Gonzalez and Alor-Hernandez [10] and Rodriguez-Gonzalez *et al.* [27] argued for the necessity of multi-level diagnosis. To demonstrate their methods, they applied several logical description rules to their model and used rule inference techniques to derive diagnostic results. However, their model was constructed manually and ignored users' demographic information, such as gender and age. Furthermore, their method does not guarantee accuracy of normal diagnosis because all of the specified rules are solely for multi-level diagnosis. Finally, despite the necessity of proving generality and validity through intensive evaluation, they showed only preliminary results for only 20 clinical cases.

Following rapid improvements in computational power, researchers have been increasingly attracted to deep learning techniques such as IBM Watson [28] to assist with health diagnosis. However, according to a survey by Ravi *et al.* [29], several challenges need to be resolved for the application of deep learning. For example, training a deep learning architecture requires an extensive amount of labeled data that must be created manually and extensive computing resources to avoid excessive consumption of computational time. Deep learning can also be affected by convergence issues and overfitting. Hence, various supplementary learning strategies are required to address these problems.

In summary, much research has been undertaken to construct enriched knowledge models and reinforce the effectiveness and reliability of health diagnosis techniques. Despite these efforts, most of the studies so far have difficulty in automatically constructing a knowledge model, and personal health attributes have not been adequately considered as an essential component of diagnostic knowledge. Furthermore, the terminology processes for abbreviations and codes remain poor, and the accuracy of diagnostic results cannot be guaranteed because the results from related studies have not been rationalized through comparison with various measurements and sufficient numbers of test cases. We fill these gaps by using our proposed techniques.

III. OVERVIEW OF THE PROPOSED TECHNIQUE

In this section, we present an overview of our proposed intelligent health diagnosis technique. Fig. 1 schematically illustrates the operation of the proposed technique through a brief scenario involving two users. Detailed explanations of the two internal components will be presented in Section IV.

To deliver final diagnostic results to a user, our intelligent health diagnosis technique requires information derived from the HDDO via semantic inferences. Therefore, we first generate the HDDO via an internal component called automatic ontology generation by exploiting three types of biomedical resources: source ontologies (i.e., DO and SYMP), BioPortal, and PubMed. Initially, we extract all disease and symptom terms contained in DO and SYMP and expand the extracted terms by acquiring terminological knowledge such as synonyms, abbreviations, and codes via BioPortal. Subsequently, we generate the HDDO, where the diseases, symptoms, and personal health attributes are defined, by using ontology components such as classes, object properties, instances, and datatype properties.

To specify the relationships between diseases and symptoms and between diseases and personal health attributes, we exploit journal article IDs and their MeSH metadata from PubMed and acquire all relationships by using term co-occurrence analysis. Specifically, in the case of the relationships between diseases and personal health attributes, we classify the diseases that require specific age, gender, and health conditions via the analysis of MeSH metadata, and their relationships are then specified. Further, during the term co-occurrence analysis, we store the frequencies of term co-occurrences between diseases and symptoms and between diseases and personal health attributes for later use in our ranking methods. Following the ontology generation, we deploy the HDDO to perform an individual user's health diagnosis.

As shown in the left of Fig. 1, when users A and B with different health records enter the same symptoms as input queries, another internal component, called a health diagnosis, automatically synchronizes with their individual PHR data in real time. Specifically, users can enter their symptoms in various forms such as synonyms, abbreviations, and codes. In the health diagnosis component, we first analyze a user's PHR data through health concept mapping procedures and then map to the user health keywords pertaining to their health conditions. Then, the user's input queries and health keywords are queried to the HDDO. Through semantic inferences of the HDDO, we recognize the various forms of user inputs and health keywords and identify possible diagnoses by filtering out irrelevant diseases. Finally, we calculate the ranking score for each disease in possible diagnoses with consideration of the importance of the disease-symptom relationships and the disease-personal health attribute relationships. In addition, to support multi-level diagnosis, we consider the similarity weights between the possible diagnoses and diseases recorded in the user's PHR data and reflect them in the ranking score for each disease. We then select the top-k ranked diseases as the final output. Therefore, as shown in Fig. 1, based on differences in their respective PHR data, the final outputs for users A and B can have different results, even if they enter the same symptoms. However, the diagnostic results provided by the proposed technique are not always accurate. Therefore, we display the percentage likelihood next to each potential disease in the diagnostic results to indicate how likely the user is to experience the particular disease.

In addition, since the symptoms used in the initial diagnosis can change over time, we provide the user's diagnostic progress information from the diagnostic result log which can be stored in the HDDO. When the user's initial diagnosis has been completed, we store all relationships used in the diagnosis in the user's diagnostic result log. If the user wants to know the how the disease is progressing, we can query the updated relationships based on symptoms added or removed by the user. We then recalculate the ranking score for the diagnosed diseases. Thus, as shown in Fig. 2, if symptoms have been added or removed based on a previous diagnosis, the ranking or percentage rate of each diagnosis will have changed in the diagnosis results. This allows users to track or monitor previously diagnosed diseases or other possible diseases.

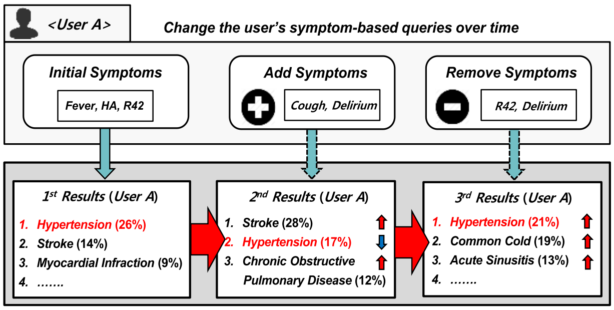


FIGURE 2. Example of user's diagnostic progress information.

IV. INTELLIGENT HEALTH DIAGNOSIS TECHNIQUE

This section presents detailed descriptions of the two internal components in the proposed techniques: automatic ontology generation and health diagnosis.

A. AUTOMATIC ONTOLOGY GENERATION COMPONENT

This component is used to automatically generate the HDDO. The HDDO is an upper-level ontology for personal health diagnosis, and it is used to identify possible diagnoses from the user's input queries and PHR data. The HDDO contains three main classes to represent classified concepts and 13 object properties to establish relations between instances of classes. Fig. 3 depicts a structural overview of the HDDO.

The *PHR* class contains information on personal health and comprises three subclasses: *User, Demographic Information,* and *Health Conditions*. The *User* subclass contains user IDs and names to identify individual users. To represent individual demographics, the *User* subclass connects with the *Demographic Information* subclass via four object properties: *hasAge*, *hasGender*, *hasPregnancy*, and *hasMenstruation*. The *User* subclass also connects with the *Health Condition* subclass to represent individual health conditions via six object properties: *hasBP*, *hasFPG*, *hasBMI*, *hasBodyTemp*, *hasPulse*, and *hasHemo*. To represent a user's medical records, the *User* subclass connects with the *Medical Records* subclass via the *hasRecords* object property. Finally, the *User* subclass connects with the *Diagnostic Logs* subclass via the *hasLogs* object property to store the user's diagnostic results. When diagnostic results are updated with additional user queries, the result logs are automatically generated as different version and stored in the *Diagnostic Logs* subclass.

The *Symptom* class is used to represent a user's symptoms and connects to the *PHR* class via the *hasSymptom* object property. The *Symptom* class contains n number of symptom subclasses that are extracted from SYMP. In each symp-

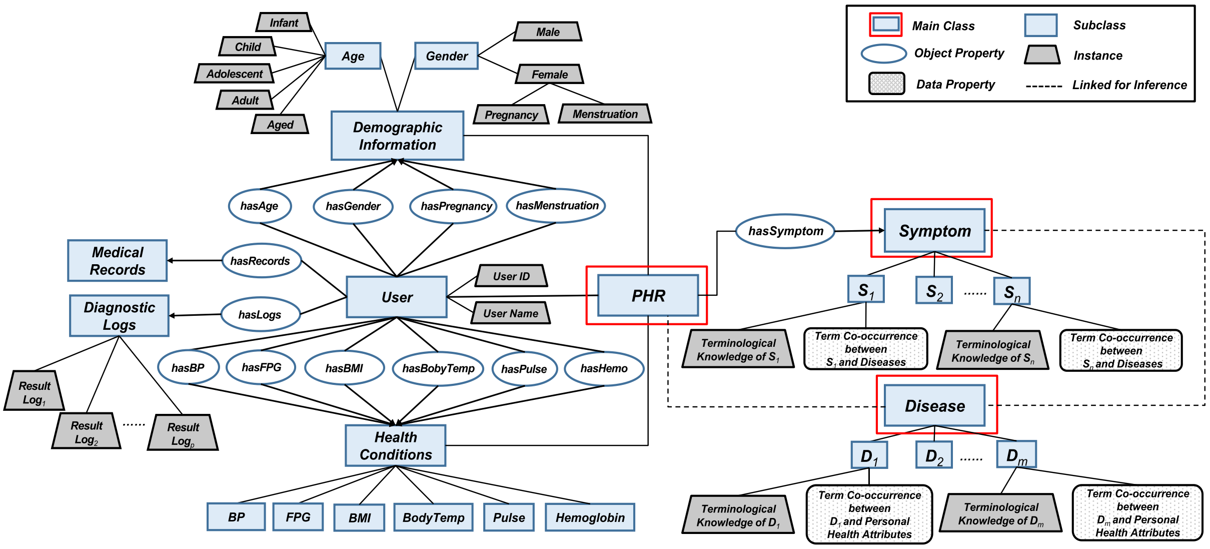


FIGURE 3. Structural overview of the Human Disease Diagnosis Ontology.

tom subclass, the terminological knowledge acquired from BioPortal is stored as ontology instances, and the frequency of term co-occurrence between diseases and symptoms analyzed from PubMed is stored as ontology data properties. The frequency of term co-occurrence between diseases and symptoms is utilized later in the health diagnosis component to calculate the importance of disease-symptom relationships.

The *Disease* class is used to identify each user's possible diagnoses based on their symptoms and PHR data. In the same manner, the Disease class contains m number of disease subclasses that are extracted from DO, and the terminological knowledge acquired from BioPortal is stored as ontology instances in each disease subclass. However, since possible diagnoses are identified via semantic inferences, the *Disease* class is connected to the *PHR* and *Symptom* classes to specify the rules used for inferences. As in the *Symptom* class, each disease subclass stores the frequency of term co-occurrence between diseases and personal health attributes based on analysis from PubMed. This frequency of term co-occurrence is utilized later in the health diagnosis component to calculate the importance of disease-personal health attribute relationships.

To automatically generate the HDDO, we designed two processing modules within this component: term extraction and expansion, and ontology generation. Fig. 4 shows the architecture of the automatic ontology generation component.

1) TERM EXTRACTION AND EXPANSION MODULE

This module initially extracts disease and symptom terms using DO and SYMP. DO and SYMP are well-established ontologies that are focused on representing human disease and symptom terms captured across biomedical resources. These two ontologies are currently adopted as standards under the Open Biological and Biomedical Ontology (OBO) Foundry, and the terms of diseases and symptoms are well-defined as ontology classes using standard references.

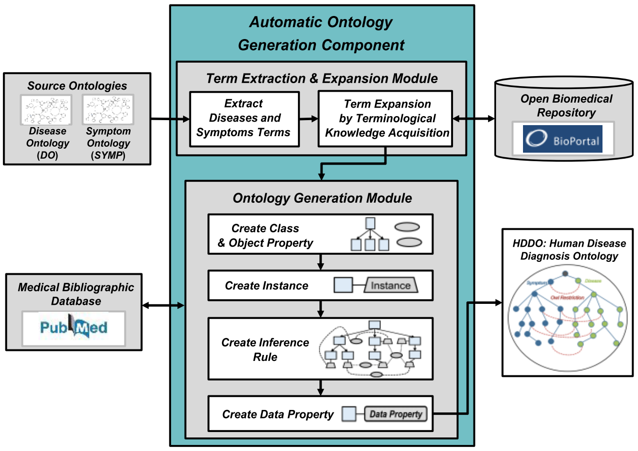
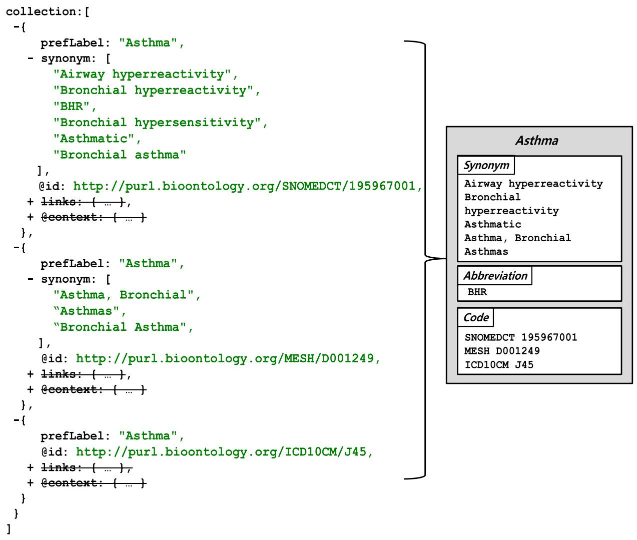

Automatic Ontology Generation Component
Source Ontologies
Disease Ontology (DO)
Symptom Ontology (SYMP)
Term Extraction & Expansion Module
Extract Diseases and Symptoms Terms
Term Expansion by Terminological Knowledge Acquisition
Open Biomedical Repository
BioPortal
Ontology Generation Module
Create Class & Object Property
Create Instance
Instance
Create Inference Rule
Create Data Property
Data Property
Medical Bibliographic Database
PubMed
HDDO: Human Disease Diagnosis Ontology

FIGURE 4. Architecture of automatic ontology generation component.

However, in the medical domain, different terms may have the same semantic meaning [30], for instance, "*neoplasm* " and "*cancer* ". Furthermore, terms can be used as syntactic variants, for instance, plurals such as "*external fistula* " and "*external fistulas* " or abbreviations such as "*COPD* " and "*chronic obstructive pulmonary disease* ". Therefore, to generate an ontology that can recognize various forms of disease and symptom terms, we expand the disease and symptom terms by acquiring terminological knowledge for variant terms stored in BioPortal and classifying them into synonyms, abbreviations, and codes. Fig. 5 shows an example of terminology expansion using BioPortal.

BioPortal is an open biomedical repository that contains multiple biomedical terminology resources and codes such as Systematized Nomenclature of Medicine-Clinical Terms (SNOMED-CT), and International Clinical



```
collection:[
 -{
    prefLabel: "Asthma",
  - synonym: [
      "Airway hyperreactivity",
      "Bronchial hyperreactivity",
      "BHR",
      "Bronchial hypersensitivity",
      "Asthmatic",
      "Bronchial asthma"
    ],
    @id: http://purl.bioontology.org/SNOMEDCT/195967001,
  + ~~links: { … }~~,
  + ~~@context: { … }~~
  },
 -{
    prefLabel: "Asthma",
  - synonym: [
      "Asthma, Bronchial",
      “Asthmas",
      “Bronchial Asthma",
    ],
    @id: http://purl.bioontology.org/MESH/D001249,
  + ~~links: { … }~~,
  + ~~@context: { … }~~
  },
 -{
    prefLabel: "Asthma",
    @id: http://purl.bioontology.org/ICD10CM/J45,
  + ~~links: { … },~~
  + ~~@context: { … }~~
  }
 }
]
```

FIGURE 5. Example of the terminology expansion via BioPortal.

TABLE 1. Example of expanded terminological knowledge obtained via bioportal.

Disease/Symptom	Categories	Terminological Knowledge
Headache (Symptom)	Synonyms	Pain in Head, Cephalgia, Head Pain, Headaches
	Abbreviation	HA
	Codes	784.0(ICD9-CM), R51(ICD10-CM), 25064002(SNOMED-CT)
Diabetes Mellitus (Disease)	Synonyms	Diabetes, Glucose Intolerance, Hyperglycemia, Glycosuria
	Abbreviation	DM
	Codes	250(ICD-9 CM), 0371(WHO) 73211009(SNOMED-CT)
Fever (Symptom)	Synonyms	Pyrexia, Hyperthermia, Temperature Elevation
	Codes	780.60(ICD9-CM), R50.9(ICD10-CM), 386661006(SNOMED-CT)
Asthma (Disease)	Synonyms	Airway Hyperreactivity, Asthmatic, Bronchial Asthma
	Abbreviation	BA, BHR
	Codes	493(ICD9-CM), J45(ICD10-CM), D001249(MeSH)

Classification of Diseases-Tenth Revision (ICD-10). BioPortal stores the terminological knowledge of multiple resources through categorization into the *prefLabel*, *synonym*, *@id*, *links*, and *@context* properties as shown in Fig. 5 and provides a search REST API[5] that facilitates knowledge discovery via user-entered text. In this paper, we focus on the *prefLabel*, *synonym*, and *@id* properties of BioPortal to classify synonyms, abbreviation, and codes of extracted diseases and symptoms.

As *prefLabel* and *synonym* represent well-written terms or abbreviations for diseases and symptoms, they are used to classify synonyms and abbreviations without text processing. In the case of code, we parse the *@id* and classify them into the code of extracted disease and symptom terms. Since BioPortal contains various types of terminological knowledge from multiple resources, duplicated or ambiguous terms (i.e., terms that have various meanings and link to many other terms) may include a set of synonyms, abbreviations, and codes. Therefore, we filter these duplicated or ambiguous terms during the term expansion. We then store the extracted disease and symptom terms with their expanded terminological knowledge and pass to the next module, ontology generation. Table 1 shows an example of the expanded terminological knowledge obtained via BioPortal.

2) ONTOLOGY GENERATION MODULE

The ontology generation module is used to generate the HDDO that specifies the relationships between diseases and symptoms and between diseases and personal health attributes based on journal article IDs and MeSH metadata from PubMed. PubMed is currently the most comprehensive literature database on biomedical sciences and uses article IDs and MeSH metadata to index articles for the purpose of facilitating literature retrieval. MeSH metadata contain lists of controlled keywords, called MeSH terms, that are used for the annotation of published articles, resulting in a high-quality representation of their main topics and contributions. Therefore, in this module, we find journal article IDs that contain MeSH terms on extracted diseases, symptoms, or personal health attributes and generate the HDDO based on all of their interrelationships obtained via term co-occurrence analysis. We generate the HDDO via four internal steps that are sequentially executed as shown in Fig. 4.

Step 1) *Create Class and Object Property:* In this step, the module creates three main classes (i.e., *PHR*, *Disease*, and *Symptom*), their subclasses, and the 13 object properties required by the HDDO. Algorithm 1 describes the procedure for creating the three main classes, and the corresponding subclasses of the *Disease* and *Symptom* classes.

The module first creates the three main classes. Since some terms for symptoms or personal health attributes (e.g., *anemia* and *hypertension*) can also be disease terms, we assign different types of ID to the main classes to uniquely identify their respective concepts. The module then adds class labels using *<rdfs:label>* to represent the name for the *PHR*, *Disease*, and *Symptom* classes.

For the *Disease* and *Symptom* classes, the HDDO does not require subclasses that cannot specify relationships via PubMed. Therefore, to create subclasses for the *Disease* and *Symptom* classes, we ensure that there are journal articles about the extracted diseases and symptom terms in PubMed. If so, the extracted terms are created as respective subclasses of the *Disease* or *Symptom* classes; otherwise, we use their terminological knowledge to ensure that PubMed journal articles exist and create subclasses if they do. For reference, to avoid finding irrelevant articles, we used a quoted search that uses double quotes in the query terms. In creating the

[5] http://data.bioontlogy.org/documentation

Algorithm 1 Create Three Main Classes, and Corresponding Subclasses for Disease and Symptom

```
Input: Extracted disease terms $D$ and symptom terms $S$,
       Terminological knowledge of the extracted disease
       terms $DTK$ and the extracted symptom terms $STK$
Output: Three main classes, and the subclasses for Disease
        and Symptom
/* Create three main classes. For unique IDs, we assigned
   PHR00, D0000, S0000, respectively. */
Create PHR, Disease, Symptom classes and assign
  a unique ID
Add class labels for PHR, Disease, Symptom classes using
  <rdfs:label>
/* Create subclasses of the Disease class. The subclasses of
   the Symptom class are created in the same manner. */
for each extracted disease term $d[i] \in D$ do
    int NumOfArticle= FindArticle ($d[i]$)
    if NumOfArticle> 0 then
        Create the subclass for $d[i]$ and assign a unique ID
          (e.g. D0001)
        Add a subclass label for $d[i]$ using <rdfs:label>
    else
        for each terminological knowledge $dtk\ [i][p] \in$
              $DTK$ do
            NumOfArticle= FindArticle ($dtk\ [i][p]$)
            if NumOfArticle> 0 then
                Create the subclass for $d[i]$ and assign a
                 unique ID
                Add a subclass label for $d[i]$ using
                 <rdfs:label>
                break
            end if
        end for
    end if
end for
```

subclasses, we assign a unique ID to each extracted disease and symptom term and add a subclass label that can represent the terms as the subclass name.

For the *PHR* class, we create subclasses by referring to the general schema structures of PHR services. To actively manage a user's health, PHR services typically comprise a user profile table, health condition table, and medical history table. Therefore, in our *PHR* class, we create *User*, *Demographic Information* and *Health Conditions* subclasses based on the user profile and health condition tables from PHR services. Specifically, to represent a user's demographic information and health conditions, we create two classes (i.e., *Age* and *Gender*) in *Demographic Information* and six classes (i.e., *BP, FPG, BMI, BodyTemp*, *Pulse,* and *Hemoglobin*) in *Health Condition* s, respectively. In addition, to represent different medical records for a given user, we create the *Medical Records* class below the *User* subclass based on the medical history table from PHR services. Finally, we create the *Diagnostic Logs* subclass below the *User* subclass to store the user's diagnostic results.

Algorithm 2 Create Ontology Instances for the Subclasses of Disease and Symptom

```
Input: Extracted disease terms $D$ and symptom terms $S$,
       Terminological knowledge of the extracted disease
       terms $DTK$ and the extracted symptom terms $STK$
Output: Ontology instances for subclasses of Disease
        and Symptom
Get $D_{sub}$ and $S_{sub}$, subclasses of the Disease and Symptom
  classes
/* Create ontology instances for subclasses of Disease class.
   The ontology instances for subclasses of Symptom class
   are created in the same manner. */
for each subclass $d_{sub}\ [i] \in D_{sub}$ do
    Create an instance $k$ for $d_{sub}[i]$ and assign a unique
     ID (e.g. DI0001)
    Get subclass label of $d_{sub}[i]$ and add it to the instance
     label of $k$
    for each disease term $d[j] \in D$ do
        if $d[j]$ =instance label of $k$ then
            for each terminological knowledge $dtk\ [j][p] \in$
                  $DTK$ do
                Create an instance $q$ for $dtk\ [j][p]$ and assign
                 a unique ID
                Add the term of $dtk\ [j][p]$ as the instance
                 label of $q$
                Link instance $q$ to $k$ using <owl:sameAs>
            end for
        end if
    end for
end for
```

After creating three main classes and their subclasses, the module creates 13 object properties that are used to connect the subclasses.

Step 2) *Create Instance:* In the create instance step, the module creates the ontology instances for storing terminological knowledge within the subclasses of the *Disease* and *Symptom* classes and for storing keywords about user demographics and health attributes within the subclasses of the *PHR* class. The ontology instances are elements of a given ontology subclass and are used to describe the concepts underlying the subclass. They are also used to specify inference rules that allow for inferring new facts and can be treated the same as other ontology instances of the ontology subclass via semantic inferences. Using these characteristics, we create ontology instances for terminological knowledge and store them into subclasses of the *Disease* and *Symptom* classes to address disease and symptom terminology issues where the homogeneity of terminology is particularly problematic. We also exploit the created ontology instances to specify the relationships between the *Disease* and *Symptom* classes or *Disease* and *PHR* classes in the next step, in which the inference rules are created. Algorithm 2 describes the

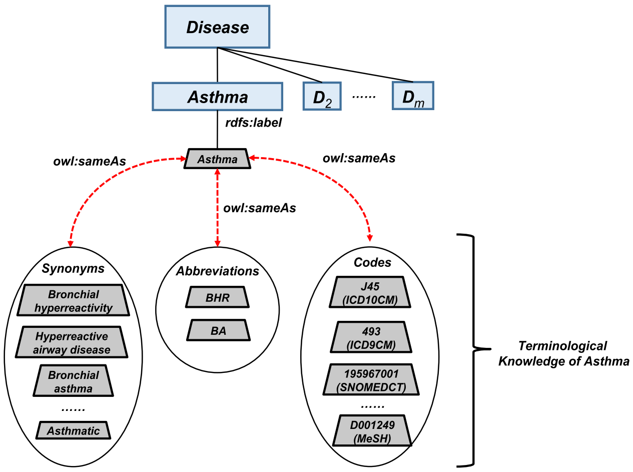


FIGURE 6. Schematic diagram illustrating the linking of ontology instances for terminological knowledge.

procedure used to create ontology instances within the subclasses of the *Disease* and *Symptom* classes.

The module first acquires all subclasses of the *Disease* and *Symptom* classes that were created in the previous step. Subsequently, the module creates an ontology instance and assigns a unique ID to each subclass of the *Disease* and *Symptom* classes. To represent a name for each created ontology instance, we obtain the subclass label of each subclass and add it to the instance label. To create ontology instances for terminological knowledge and store them into each subclass,

the module obtains terminological knowledge for each created ontology instance. The module then creates an ontology instance and unique ID for each terminological knowledge and adds an ontology instance label. Finally, the module links the ontology instances for the terminological knowledge to the ontology instance for each subclass of the *Disease* or *Symptom* class using the <*owl:sameAS*> statements. Terminological knowledge linked by these <*owl:sameAS*> statements can be treated as equivalent ontology instances via semantic inferences even if the symptoms input by users or diseases in their PHR data are variously represented via synonyms, abbreviations, and codes. Fig. 6 shows a schematic diagram that illustrates the linking of ontology instances for terminological knowledge via <*owl:sameAS*> statements.

After creating instances for subclasses of the *Disease* and *Symptom* classes, the module creates instances for the subclasses of the *PHR* class. Among the subclasses of the *PHR* class, we only create ontology instances for the *Demographic Information* and *Health Conditions* subclasses that are required to specify relationships with diseases in the next step. In the case of the *User* and *Medical Records* subclasses, we automatically store the user's login information and medical histories of their PHR data as ontology instances upon the initial user query to the system. For the *Demographic Information* subclass, the module creates ontology instances for the *Age* and *Gender* subclasses. Because these ontology instances should be capable of specifying relationships with diseases through information extracted from PubMed in the next step, we create the ontology instances by selecting age- and gender-related MeSH terms as the demographic keywords. For the *Health Conditions* subclass, the module creates ontology instances for the *BP*, *FPG*, *BMI*, *BodyTemp*, *Pulse,* and *Hemoglobin* subclasses that represent the user's health condition; these are also created by selecting relevant MeSH terms based on classified health keywords found in well-known health measures. Table 2 shows the ontology instances created for the *Demographic Information* and *Health Conditions* subclasses.

Step 3) Create Inference Rule: In this step, the module creates inference rules for the relationships between diseases and symptoms and between diseases and personal health attributes. The inference rules are created using description logic comprising ontology instances and object properties and are specified in each subclass of the *Disease* class using <*owl:equivalentClass*>. The specified rules are used to filter out diseases that are irrelevant to users and to provide possible diagnoses via semantic inferences from the HDDO. The inference rules can be defined as follows:

$$\begin{aligned} DIS\_X \equiv\ & \exists\ hasSymptom(oneof\{syp_1, syp_2, .., syp_n\}) \sqcap \\ & \exists\ hasAge.\{age\} \sqcap hasGender.\{gender\} \sqcap \\ & \exists\ hasPregnancy.\{pregnancy\} \sqcap \ldots \sqcap \\ & \exists\ hasBP.\{bp\} \sqcap \\ & \exists\ hasFPG.\{fpg\} \sqcap \\ & \exists\ hasBMI.\{bmi\} \sqcap \ldots \sqcap hasSymptom = n \end{aligned} \tag{1}$$

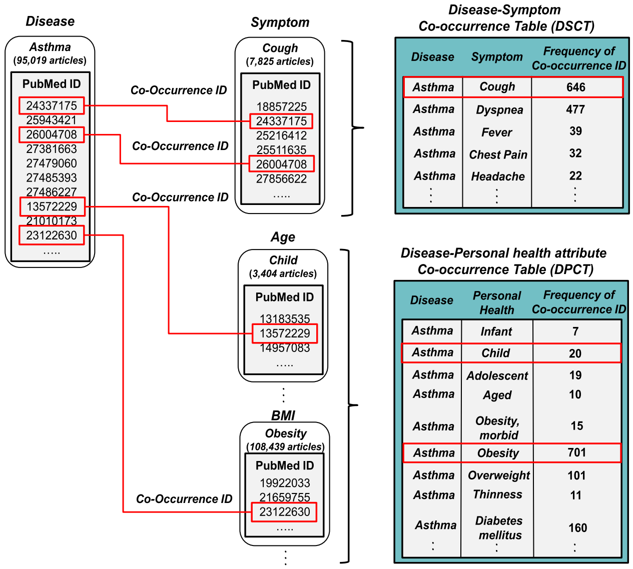
where *DIS_X* is a subclass of the *Disease* class, and the following description is the rule for *DIS_X*. This rule indicates that a user has *DIS_X* if they have at least one of the given symptoms (e.g., $syp_1, syp_2, \ldots, syp_n$) and the precise demographic characteristics and health conditions associated with *DIS_X*. The values contained in the object properties (syp_n, *age*, *gender*, *pregnancy*, etc.) are all ontology instances. The number of symptoms of *DIS_X* are specified at the end of the rule to establish cardinalities for semantic inferences; these are used to filter*DIS_X* if the user queries more symptoms than are contained in *DIS_X*. Finally, the rule is specified by linking to *DIS_X* using <*owl:equivalentClass*>. For example, the rule for gestational diabetes, which is a specific type of diabetes that occurs during pregnancy, is defined as follows:

$$\begin{aligned} & GestationalDiabetes \\ & \quad \equiv \exists\ hasSymptom(oneof\{vomiting, albuminuria, \\ & \qquad sleepdeprivation, \ldots, polyuria\}) \sqcap \\ & \quad \exists\ hasAge.\{adult\} \sqcap \exists hasGender.\{female\} \sqcap \\ & \quad \exists\ hasPregnancy.\{pregnancy\} \sqcap \\ & \quad \exists\ hasFPG.\{diabetesmellitus\} \sqcap hasSymptom = 37 \end{aligned}$$

To create these inference rules, it is first necessary to know what symptoms and personal health attributes are associated

TABLE 2. The Number of diagnosis cases used in the performance evaluation.

Demographic/Health Condition Subclasses	Ontology Instances
Age	"Infant", "Child", "Adolescent", "Adult", "Aged"
Gender	"Male", "Female", "Pregnancy", "Menstruation"
BP	"Hypotension", "Prehypertension", "Normal", "Hypertension", "Hypertension, Malignant"
FPG	"Hypoglycemia", "Normal", "Prediabetic State", "Diabetes Mellitus"
BMI	"Underweight", "Normal", "Overweight", "Obesity", "Obesity, Morbid"
BodyTemp	"Hypothermia", "Normal", "Fever", "Hyperthermia Malignant"
Pulse	"Bradycardia", "Normal", "Tachycardia"
Hemoglobin	"Anemia", "Normal", "Polycythemia"



Disease-Symptom Co-occurrence Table (DSCT)

Disease	Symptom	Frequency of Co-occurrence ID
Asthma	Cough	646
Asthma	Dyspnea	477
Asthma	Fever	39
Asthma	Chest Pain	32
Asthma	Headache	22
⋮	⋮	⋮

Disease-Personal health attribute Co-occurrence Table (DPCT)

Disease	Personal Health	Frequency of Co-occurrence ID
Asthma	Infant	7
Asthma	Child	20
Asthma	Adolescent	19
Asthma	Aged	10
Asthma	Obesity, morbid	15
Asthma	Obesity	701
Asthma	Overweight	101
Asthma	Thinness	11
Asthma	Diabetes mellitus	160
⋮	⋮	⋮

FIGURE 7. Example of finding the relationships between diseases and symptoms and between diseases and personal health attributes.

with each disease. Since the full-text analysis methods do not yield results with comparable accuracy, as was proven by Zhou *et al.* [22], the module finds relationships between diseases and symptoms and between diseases and personal health attributes using PubMed journal article IDs and their related MeSH terms. Fig. 7 shows an example of finding these relationships.

Initially, we query PubMed using the terms of the subclasses of the *Disease* and *Symptom* classes to obtain their related journal article IDs. We then find the relationships between diseases and symptoms using term co-occurrence analysis to analyze PubMed journal articles in which two IDs appear together. The acquired relationships and frequencies of the co-occurrence IDs are then stored in the Disease-Symptom Co-occurrence Table (DSCT) and used to create inference rules. In the same manner, the relationships between diseases and personal health attributes are found by querying the ontology instances created for the subclasses of the *PHR* class, and the acquired relationships and frequencies of the co-occurrence IDs are then stored in the Disease-Personal health attributes Co-occurrence Table (DPCT). However, unlike the relationships between diseases and symptoms, there are a number of problems in applying relationships between diseases and personal health attributes directly to the inference rules. For example, as shown in Fig. 7., we can know that asthma has strong relationships with being a child and obesity because there are high frequencies for both terms in the DPCT. If the inference rules are defined by these relationships, asthma can be provided as a possible diagnose through semantic inference only when the user is a child and their BMI indicates obesity. It is well-known, however, that asthma is a disease that can occur under other health conditions. To address these problems, we classify the diseases in the DPCT as age- and gender-specific diseases and the diseases that are related to health conditions (i.e., *BP*, *FPG*, *BMI*, *Body Temperature*, *Pulse*, and *Hemoglobin*).

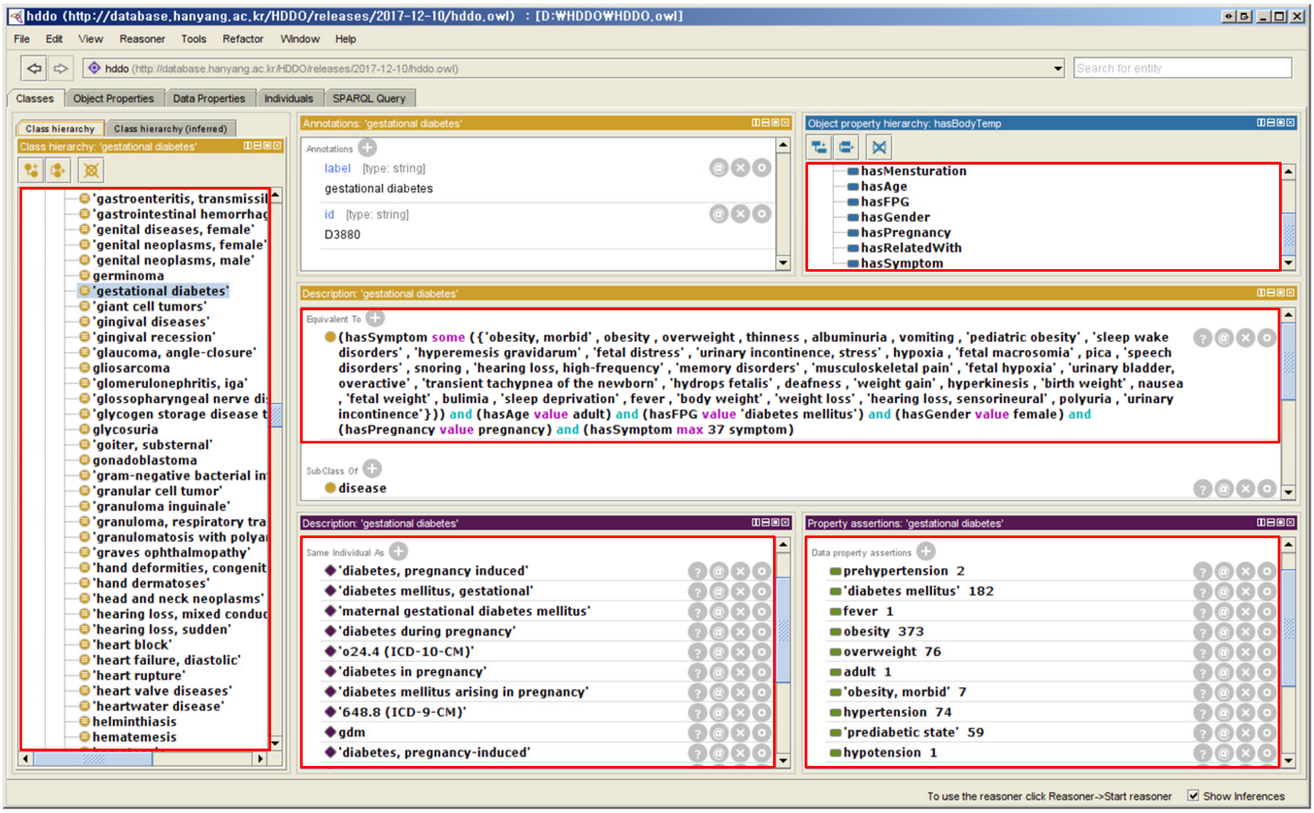
To classify age- and gender-specific diseases, we exploit the hierarchical structure of the MeSH terms presented in the MeSH metadata. In this metadata, the terms are arranged hierarchically with 16 top nodes representing categories such as " *anatomy* ", " *diseases* ", " *phenomena and processes* ", etc. The category of " *diseases* " contains diseases that occur in specific age groups such as neonatal sepsis and juvenile myoclonic epilepsy under age-represented subcategories (e.g., " *Infant, Newborn, Disease* " subcategory). Therefore, we extract diseases from age-represented subcategories and classify them by matching with the diseases in the DPCT. The hierarchical structure of the MeSH terms also includes subcategories such as " *Male Urogenital Disease*" and " *Female Urogenital Diseases and Pregnancy Compilations*" that describe gender-specific diseases. Thus, we extract diseases from their respective subcategories and classify them by matching them with the diseases in the DPCT. For the diseases related to health conditions, we classify them using the text descriptions for MeSH terms. The MeSH metadata contain text descriptions that provide detailed explanations of each disease term. Thus, we check whether the ontology instances for the subclasses of the PHR classes are contained in a given disease text description through syntax matching; if so, the corresponding disease is classified as a disease that is related to a health condition.

Algorithm 3 Create Inference Rules

```
Input: Disease-Symptom Co-occurrence Table DSCT, Age-specific diseases $D_{age}$, Gender-specific diseases $D_{gender}$, and
       Diseases that related to health conditions $D_{healthCond}$
Output: Inference rules for each subclass of Disease Class
Get $D_{sub}$, $S_{sub}$, and $P_{sub}$, subclasses of the Disease, Symptom, and PHR classes
/* Create an inference rule for each subclass of Disease class */
for each subclass $d_{sub}[i] \in D_{sub}$ do
    Get the subclass label of $d_{sub}[i]$
    Create EnumeratedClass for symptoms
/* Create the rule for symptoms*/
    for each relation $rel\ [k] \in DSCT$ do
        if subclass label of $d_{sub}[i] =$ disease in $rel\ [k]$
            Obtain the symptom pfrom $rel\ [k]$
            Find the ontology instance ID for symptom $p$ from $S_{sub}$, and add it to the EnumeratedClass using < owl:oneOf>
        end if
    end for
    Create < owl:someValuesFrom> restriction, and add EnumeratedClass to this restriction using hasSymptom object
     property
/* Create the rule for personal health attributes (i.e., age, gender, health conditions) */
/* Below is the rule for age. The rule for gender is created in the same manner. */
    for each disease $d[k] \in D_{age}$do
        if $d[k] =$ subclass label of $d_{sub}[i]$
            Get the age of $d[k]$ from $D_{age}$, and find the ontology instance ID for the age from $P_{sub}$
            Create < owl:hasValue> restriction, and add an ontology instance ID to this restriction using hasGender
             object property
        end if
    end for
/* Below is the rule for health conditions (e.g. BP) */
for each disease $d[l] \in D_{healthCond}$ do
    if $d[l] =$ subclass label of $d_{sub}[i]$
        Get the health condition of $d[l]$ and find the ontology instance ID from $P_{sub}$
        if ontology instance ID contained in the BP subclass
            Create < owl:hasValue> restriction, and add health condition into this restriction using hasBP object property
        else if ontology instance ID contained in the FPG subclass
/* The rules for FPG, BMI, BodyTemp, Pulse, and Hemoglobin are created in the same manner with the rule for BP*/
        end if
    end if
end for
/* Create the rule for cardinalities of symptoms*/
    Get the number of symptoms $S_{num}$ from the EnumeratedClass
    Create < owl:maxQualifiedCardinality> restriction and add $S_{num}$ using hasSymptom object property
/* Combine the rules for symptoms, personal health attributes, and cardinalities of symptoms using < owl:intersectionOf> */
    Create the list $L_{Rules}$ for the rules for symptoms, personal health attributes, and cardinalities of symptoms
    Create < owl:intersectionOf> property and add $L_{Rules}$ into this property.
/* Specify the rule using < owl:equivalnentClass> */
    Create < owl:equivalnentClass> and add < owl:intersectionOf> property to specify the rule for $d_{sub}[i]$
end for
```

Following this, we create the inference rules using the disease-symptom relationships stored in the DSCT, the age- and gender-specific diseases, and the diseases related to health conditions. Algorithm 3 shows the procedure for creating the inference rules for each subclass of the *Disease* class. Roughly speaking, the algorithm first creates the rules related to symptoms, personal health attributes, and cardinalities for each subclass of the *Disease* class and then combines the created rules using < *owl: intersectionOf*> and specifies these combined rules to each subclass of the *Disease* class using < *owl: equivalentClass*>.

To create the inference rules, the module first gets the subclasses of the *Disease*, *Symptom,* and *PHR* classes. Then, it obtains a label for each subclass of the *Disease*



hddo (http://database.hanyang.ac.kr/HDDO/releases/2017-12-10/hddo.owl) : [D:₩HDDO₩HDDO.owl]

hddo (http://database.hanyang.ac.kr/HDDO/releases/2017-12-10/hddo.owl)

Class hierarchy: 'gestational diabetes'

'gastroenteritis, transmissil
'gastrointestinal hemorrhag
'genital diseases, female'
'genital neoplasms, female'
'genital neoplasms, male'
germinoma
'gestational diabetes'
'giant cell tumors'
'gingival diseases'
'gingival recession'
'glaucoma, angle-closure'
gliosarcoma
'glomerulonephritis, iga'
'glossopharyngeal nerve di
'glycogen storage disease t
glycosuria
'goiter, substernal'
gonadoblastoma
'gram-negative bacterial in
'granular cell tumor'
'granuloma inguinale'
'granuloma, respiratory tra
'granulomatosis with polya
'graves ophthalmopathy'
'hand deformities, congenit
'hand dermatoses'
'head and neck neoplasms'
'hearing loss, mixed conduc
'hearing loss, sudden'
'heart block'
'heart failure, diastolic'
'heart rupture'
'heart valve diseases'
'heartwater disease'
helminthiasis
hematemesis

Annotations: 'gestational diabetes'

label [type: string]
gestational diabetes

id [type: string]
D3880

Object property hierarchy: hasBodyTemp

hasMensturation
hasAge
hasFPG
hasGender
hasPregnancy
hasRelatedWith
hasSymptom

Description: 'gestational diabetes'

Equivalent To

(hasSymptom some ({'obesity, morbid' , obesity , overweight , thinness , albuminuria , vomiting , 'pediatric obesity' , 'sleep wake disorders' , 'hyperemesis gravidarum' , 'fetal distress' , 'urinary incontinence, stress' , hypoxia , 'fetal macrosomia' , pica , 'speech disorders' , snoring , 'hearing loss, high-frequency' , 'memory disorders' , 'musculoskeletal pain' , 'fetal hypoxia' , 'urinary bladder, overactive' , 'transient tachypnea of the newborn' , 'hydrops fetalis' , deafness , 'weight gain' , hyperkinesis , 'birth weight' , nausea , 'fetal weight' , bulimia , 'sleep deprivation' , fever , 'body weight' , 'weight loss' , 'hearing loss, sensorineural' , polyuria , 'urinary incontinence'})) and (hasAge value adult) and (hasFPG value 'diabetes mellitus') and (hasGender value female) and (hasPregnancy value pregnancy) and (hasSymptom max 37 symptom)

SubClass Of

disease

Description: 'gestational diabetes'

Same Individual As

'diabetes, pregnancy induced'
'diabetes mellitus, gestational'
'maternal gestational diabetes mellitus'
'diabetes during pregnancy'
'o24.4 (ICD-10-CM)'
'diabetes in pregnancy'
'diabetes mellitus arising in pregnancy'
'648.8 (ICD-9-CM)'
gdm
'diabetes, pregnancy-induced'

Property assertions: 'gestational diabetes'

Data property assertions

prehypertension	2
'diabetes mellitus'	182
fever	1
obesity	373
overweight	76
adult	1
'obesity, morbid'	7
hypertension	74
'prediabetic state'	59
hypotension	1

To use the reasoner click Reasoner->Start reasoner ☑ Show Inferences

FIGURE 8. Screenshot of the Human Disease Diagnosis Ontology.

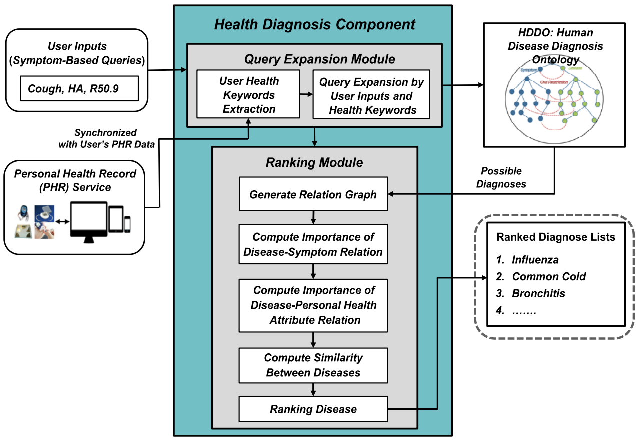

Health Diagnosis Component
User Inputs
(Symptom-Based Queries)
Cough, HA, R50.9
Query Expansion Module
User Health Keywords Extraction
Query Expansion by User Inputs and Health Keywords
HDDO: Human Disease Diagnosis Ontology
Symptom
Disease
Owl Restriction
Synchronized with User's PHR Data
Personal Health Record (PHR) Service
Ranking Module
Generate Relation Graph
Compute Importance of Disease-Symptom Relation
Compute Importance of Disease-Personal Health Attribute Relation
Compute Similarity Between Diseases
Ranking Disease
Possible Diagnoses
Ranked Diagnose Lists
1. Influenza
2. Common Cold
3. Bronchitis
4. ....

FIGURE 9. Architecture of the health diagnosis component.

class and creates an *EnumeratedClass* to contain a set of symptoms. To create the rule for symptoms, the module obtains the related symptoms in the DSCT using the subclass label and finds the ontology instance IDs of the symptoms in the subclass of the *Symptom* class. The module then adds the ontology instance IDs to the *EnumeratedClass* using < *owl:oneOf*>; this allows inference of a disease if the user has at least one given symptom. The module also creates a restriction called < *owl:someValuesFrom*> and adds the *EnumeratedClass* using the *hasSymptom* object property to the < *owl:someValuesFrom*> restriction.

After creating the rule for the symptoms, the module creates rules for personal health attributes; specifically, the module creates each of the rules for diseases related to age, gender, and health conditions. To create a rule for age, the module obtains the related disease in the list of classified age-specific diseases using the subclass labels of the *Disease* class. If the subclass labels are contained in the classified age-specific diseases list, the module gets the age from the classified lists and finds the ontology instance ID from the subclass of the *PHR* class.

The module then creates the <*owl:hasValue*> restriction and adds the ontology instance ID for the age to this restriction using the *hasAge* object property. Unlike <*owl:oneOf*>, <*owl:hasValue*> can infer a disease only if the user is the exact age defined in the rule. In the same manner, the module creates a rule for gender using the classified gender-specific disease list and the rule specified by the *hasGender* object property. We then create the rules for health conditions under the subclass of the *Disease* class. To do this, the module obtains diseases using the subclass labels of the *Disease* class from the list of classified diseases that are related to health conditions and creates the rules for *BP*, *FPG*, *BMI*, *Body Temperature*, *Pulse*, and *Hemoglobin*. If the subclass labels are contained in the list of classified diseases, the mod-

Algorithm 4 Create Datatype Properties

Input: Disease-Symptom Co-occurrence Table *DSCT* and Disease-Personal health attribute Co-occurrence Table *DPCT*

Output: Datatype properties where the term co-occurrence frequencies of the DSCT and DPCT are stored

```
Get $D_{sub}$ and $P_{sub}$, the subclasses of the Disease and PHR
   classes
   /* Create datatype properties */
Combine $D_{sub}$ and $P_{sub}$ as the list $L$
for each subclass in the list $sub[i] \in L$ do
    Create the datatype property $dp$ for $sub[i]$ and
     assign a unique ID (e.g. DTD0001)
    Get a subclass label of $sub[i]$ from $D_{sub}$or $P_{sub}$, and
     add it to the datatype property label
end for
/* Store the frequencies of co-occurrence ID of the DSCT
and DPCT*/
for each datatype property $dp[j] \in DP$ do
    Get the datatype property label of $dp[j]$
     /* Store the frequency of co-occurrence ID of the
      DSCT */
    for each relation $rel[k] \in DSCT$ do
        if label of $dp[j]$ = disease in $rel[k]$
            Obtain symptom $p$ from $rel[k]$ and find the
             subclass for symptom $p$ from $S_{sub}$
            Obtain the term co-occurrence frequency of
             $rel[k]$ from $DSCT$
            Store $dp[j]$ and frequency into the subclass of
             $S_{sub}$
        end if
    end for
     /* Store the frequency of co-occurrence ID of the
     DPCT */
    for each relation $rel[l] \in DPCT$ do
        if label of $dp[j]$ = personal health attributes in
          $rel[l]$
            Obtain disease $q$ from $rel[l]$ and find the
             subclass for disease $q$ from $D_{sub}$
            Obtain the term co-occurrence frequency of
             $rel[k]$ from $DPCT$
            Store $dp[j]$ and frequency into the subclass of
             $D_{sub}$
        end if
    end for
end for
```

ule creates the *<owl:hasValue>* restriction and adds the ontology instance ID to the *<owl:hasValue>* restriction using the *hasBP, hasFPG, hasBMI, hasBodyTemp, hasPulse*, and *hasHemo* object properties.

To create a rule for the cardinalities of the symptom, the module retrieves the number of symptoms from the *EnumeratedClass*, and then creates a restriction called an *<owl:maxQualifiedCardinality>* and adds the number of symptoms into this restriction.

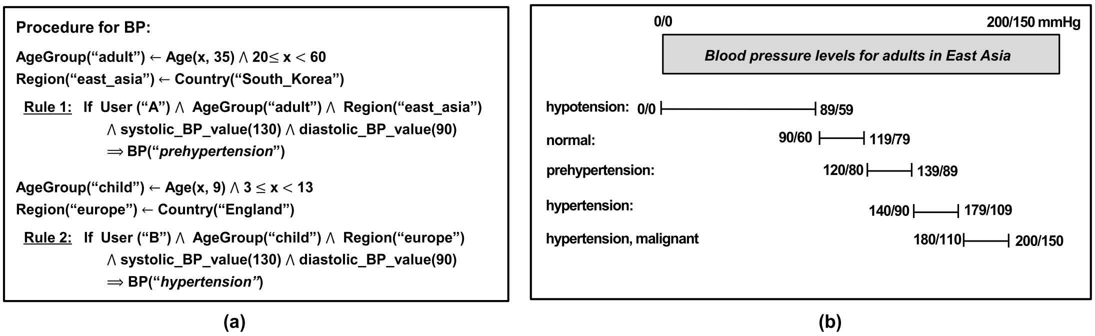
To combine the created rules for symptoms, personal health attributes, and cardinalities of symptoms, we create a list that is the container for collecting the created rules. Then, the module creates the *<owl:intersectionOf>* property and adds the list to this property. Finally, the module adds the *<owl:intersectionOf>* property to the *<owl:equivalent class>* to specify the rule for each subclass of the *Disease* class.

Step 4) *Create Datatype Property:* In this step, the module creates the datatype properties in the HDDO and stores the co-occurrence ID frequencies of the DSCT and DPCT into the subclasses of the *Symptom* and *Disease* classes, respectively. Algorithm 4 shows the procedure for creating the datatype properties and storing the co-occurrence ID frequencies of the DSCT and DPCT.

The module first gets all the subclasses of the *Disease* and *PHR* classes and combines them into a single list. It then creates the datatype properties associated with the list and assigns a unique ID to each datatype property. To represent a name for each created datatype property, the module gets the subclass label of the *Disease* and *PHR* classes and adds it to the datatype property label.

The module then stores the co-occurrence ID frequencies of the DSCT and DPCT using the created datatype properties. To store the co-occurrence ID frequency of the DSCT, the module first looks for the disease that is the same in the label of the datatype properties and in the relationships of the DSCT. The module then obtains the symptoms from the relationships of the DSCT and finds their subclasses from the *Symptom* class. The module obtains the co-occurrence ID frequency from the DSCT and stores the datatype property and the frequencies into the corresponding subclass of the *Symptom* class. The procedure for storing the co-occurrence ID frequency of the DPCT is performed similarly. The module looks for the personal health attribute that is the same in the label of datatype properties and in the relationships of the DPCT. The module then obtains the diseases from the relationships and finds the corresponding subclass from the *Disease* class. After that, the frequency of co-occurrence ID is obtained from the DPCT and stores it along with the personal health attribute in the corresponding subclass of the *Disease* class.

After completing the four internal steps, the HDDO is generated and deployed in the Web Ontology Language (OWL) format. The HDDO generated in our study contained 3,249 diseases, 295 symptoms, and totals of 5,026 and 674 pieces of terminological knowledge on diseases and symptoms, respectively. There were also 203,289 specified relations between diseases and symptoms and 84,486 specified relations between diseases and personal health attributes. Fig. 8 shows a screenshot of the automatically generated HDDO, which is displayed using the Proté gé ontology editor.



Procedure for BP:

AgeGroup(“adult”) ← Age(x, 35) ∧ 20≤ x < 60
Region(“east_asia”) ← Country(“South_Korea”)

Rule 1: If User (“A”) ∧ AgeGroup(“adult”) ∧ Region(“east_asia”)
∧ systolic_BP_value(130) ∧ diastolic_BP_value(90)
⇒ BP(“*prehypertension*”)

AgeGroup(“child”) ← Age(x, 9) ∧ 3 ≤ x < 13
Region(“europe”) ← Country(“England”)

Rule 2: If User (“B”) ∧ AgeGroup(“child”) ∧ Region(“europe”)
∧ systolic_BP_value(130) ∧ diastolic_BP_value(90)
⇒ BP(“*hypertension”*)

FIGURE 10. **Health concept mapping procedure and health indicator value for extracting user health keywords.**

B. HEALTH DIAGNOSIS COMPONENT

The health diagnosis component is used to provide personalized diagnostic results by exploiting PHR services and the HDDO. This component comprises two processing modules: query expansion and ranking. Fig. 9 shows the architecture of the health diagnosis component.

1) QUERY EXPANSION MODULE

This module prepares and executes queries to obtain possible diagnoses from the HDDO. When a user enters symptom-based queries, the module automatically synchronizes with the user’s PHR data. To query personalized health attributes together with user-entered queries, we extract user health keywords from synchronized PHR data as shown in Fig. 9.

However, most of the data in PHR services are only stored in the form of numeric values representing measured personal health conditions (e.g., *kg* for weight, *mg/dl* for blood glucose, and *mmHg* for blood pressure). Therefore, we analyze the numeric value of the user’s personal health conditions and map them to the semi-concept for health conditions through our health concept mapping procedures. The health concept mapping procedure exploits well-known measures of health (e.g., BMI, FPG, BP, body temperature, and pulse) together with the user’s demographic information (e.g., age, gender, and country region). This procedure converts the range of the measured values of the user health conditions to the appropriate keyword based on the user’s demographic information and health indicator values. Fig. 10 shows an example of the health concept mapping procedures and health indicator values for identifying the user’s BP health condition.

In the health concept mapping procedures of BP, each value for systolic and diastolic BP is mapped to an appropriate range of values with corresponding keywords by exploiting health indicator values. The health indicator values represent distributions of systolic and diastolic BP ranges according to age group and country region. As shown in Fig. 10(a), if user A is a 35-year-old South Korean, and presents with a BP of 130/85 *mmHg*, the module maps these systolic and diastolic BP values to the keyword “*prehypertension* ” using rule 1. Otherwise, if user B is a British 9-year-old, and presents the same numerical values of BP as those of user A, the module maps to the keyword “*hypertension* ” using rule 2. There are five mapping keywords for BP that coincide with the ontology instances in the *BP* class of the HDDO.

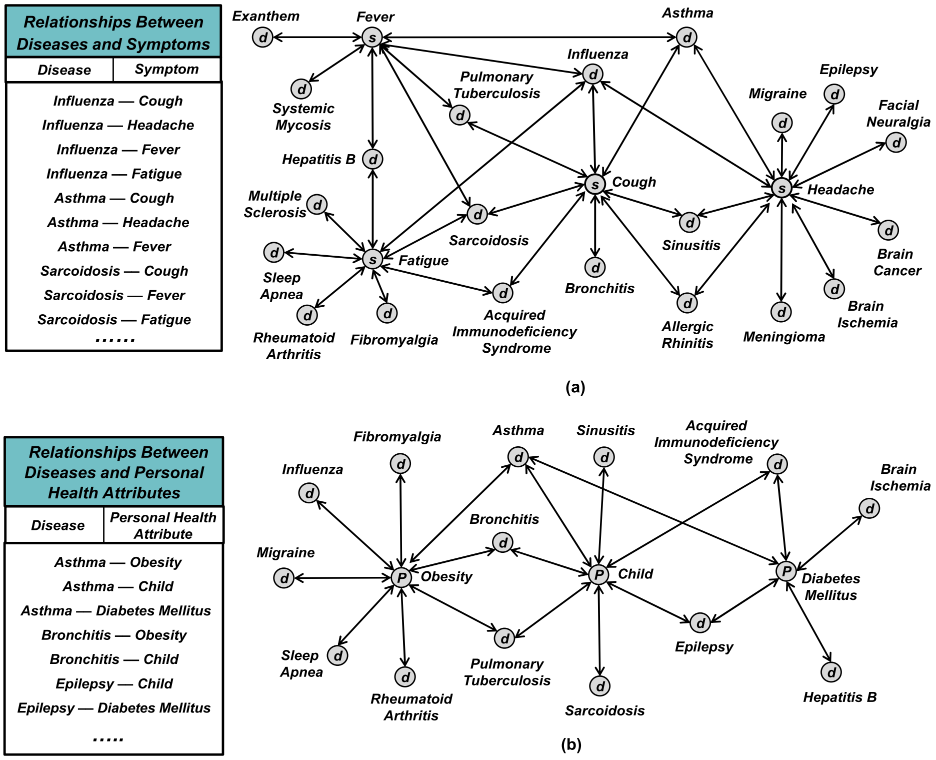
In the same manner, the health concept mapping procedures of BMI, FPG, body temperature, pulse, and hemoglobin are mapped by exploiting the health indicator values to keywords coinciding with ontology instances in the *BP, FPG, BMI, BodyTemp, Pulse*, and *Hemoglobin* subclasses of the HDDO, respectively.

Once this has been completed, the module generates a set of symptom-based queries Q (i.e., $Q = \{s_1, s_2, \ldots, s_n\}$ where n is the number of symptoms) and a set of user health keywords K that is defined as follows.

$$\begin{aligned} K &= \{hc, di, mr\}, \\ hc &= \{bmi, fpg, bp, bt, pulse, hemo\} \\ di &= \{age, gender\} \\ mr &= \{d_1, d_2, \ldots, d_p\}, 0 < p \leq 20) \end{aligned} \tag{2}$$

- K contains a set of mapped health conditions hc, a set of user’s demographic information di, and a set of medical records mr
- hc contains each keyword on BMI, FPG, BP, body temperature, pulse, and hemoglobin (e.g., $hc = \{obesity, prediabetes, prehypertension\}$)
- di contains the user’s age and gender information (e.g., $di = \{adults, male\}$)
- mr contains p number of the user’s chronic diseases or currently suffering diseases (e.g.$mr = \{bronchiectasis, emphysema, \ldots, pneumonia\}$)

The module then expands the user’s symptom-based queries Q with a set of user health keyword K and executes the queries by delivering them to the HDDO. Through semantic inference, the HDDO identifies various forms of symptoms or diseases that are included in the expanded queries and provides possible diagnoses. To perform semantic inference, we used the TrOWL reasoner, which supports the reasoning process for large ontologies and provides the fastest performance among reasoners that support description logic [31]. The HDDO then delivers a set of possible diagnoses D, (i.e., $D = \{d_1, d_2, \ldots, d_m\}$ where m is the number of diseases) to the ranking module.



Relationships Between Diseases and Symptoms

Disease	Symptom
Influenza	Cough
Influenza	Headache
Influenza	Fever
Influenza	Fatigue
Asthma	Cough
Asthma	Headache
Asthma	Fever
Sarcoidosis	Cough
Sarcoidosis	Fever
Sarcoidosis	Fatigue
......	

Relationships Between Diseases and Personal Health Attributes

Disease	Personal Health Attribute
Asthma	Obesity
Asthma	Child
Asthma	Diabetes Mellitus
Bronchitis	Obesity
Bronchitis	Child
Epilepsy	Child
Epilepsy	Diabetes Mellitus
.....	

FIGURE 11. Disease-symptom relation and disease-personal health attribute relation graphs.

2) RANKING MODULE

This module is used to calculate each disease score within possible diagnoses D and to rank each disease based on the score to provide finalized diagnostic results to a user. To calculate the score of a disease, we exploit the co-occurrence ID frequencies of the relationship between diseases and symptoms and between diseases and personal health attributes, that are stored in the HDDO. However, since these frequencies are based on the absolute number of co-occurrences of article ID in PubMed, they can lead to publication bias. For example, most biomedical researchers prioritize emergency care diseases, which are more life-threatening, or numerous symptoms that appear in various diseases. Thus, the number of articles in PubMed is biased toward abundant symptoms or emergency diseases, such as pain or breast cancer rather than specific symptoms or self-treatment diseases, such as abdominal pain or hordeolum. This means that if we used only the co-occurrence ID frequencies to rank possible diseases, emergency diseases with numerous symptoms would be prioritized over self-treatment diseases with specific symptoms. To address these heterogeneities, we do not directly use co-occurrence ID frequencies to calculate the score of a disease, but we instead use the importance of disease-symptom relationships, the importance of disease-personal health attribute relationships, and the similarity between diseases for each possible disease.

To calculate the score of a disease, the module first generates two relation graphs for possible diagnoses D, as shown in Fig. 11; one is the disease-symptom relation graph, and the other is the disease-personal health attribute relation graph. The graphs are respectively generated based on the relationships between diseases and symptoms and between diseases and personal health attributes stored in the HDDO. The disease-symptom relation graph $G = (V, E)$ is a directed graph, where V is a vertex set whose elements are each disease within possible diagnoses D and the symptoms in the user's symptom-based queries Q, and E is an edge set whose elements represent the relationships between diseases and symptoms. The disease-personal health attributes graph $G' = (V', E')$ is also a directed graph, where each disease within possible diagnoses D and its related personal health attributes constitute a vertex set V', and the relationship between diseases and personal health attributes are an edge set E'.

Subsequently, the module calculates the importance of the disease-symptom relationships for each disease in the disease-symptom relation graph G by combining PageRank and modified TF-IDF. PageRank is a widely used link-based algorithm that estimates the importance of web pages by calculating the relationship between inbound and outbound links [32]. The fundamental assumption of PageRank is that more important pages have more inbound links. In a similar manner, we assume that diseases with more inbound links are more important than others. To use PageRank, we consider each disease in possible diagnoses D and the symptoms in the user's symptom-based queries Q as web pages, and the directed edges as inbound or outbound links. The modified TF-IDF assigns a higher weight when the co-occurrence ID frequency between a disease and symptom is high, or if the disease is linked to a symptom that is rarely associated with other diseases. Otherwise, it assigns a lower weight when the co-occurrence ID frequency between a disease and symptom is low, or if the disease is linked to a symptom that is associated with many other diseases. In our case, we calculate the weight of the relationship between disease and symptom by using modified TF-IDF and calculate the importance of each disease based on the symptoms they link to through the PageRank algorithm. The importance of the disease-symptom relation for a specific disease d_m within possible diagnoses D is calculated as follows.

$$DS(d_m) = (1-d) + d \cdot \sum\nolimits_{(s_j \in In(d_m))} TF(d_m, s_j) \frac{DS(s_j)}{(|Out(s_j)|)}$$
$$TF(d_m, s_j) = \frac{Fq(d_m, s_j)}{(Total(d_m))} \cdot log \frac{N}{n_{(s_j)}}) \tag{3}$$

- d is a damping factor
- TF $(d_m,\ s_j)$ is the weight of the relationship between disease d_m and symptom s_j calculated by the modified TF-IDF
- DS (s_j) signifies the importance of the disease-symptom relations for symptom s_j, which has a relation to disease d_m
- Out (s_j) is a set of vertices to which symptom s_j points
- Fq $(d_m,\ s_j)$ is the co-occurrence ID frequency between disease d_m and symptom s_j
- $Total$ (d_m) is the total co-occurrence ID frequency of the relationships between diseases d_m and symptoms in the whole relationships
- Ndenotes the total number of diseases in the whole relationships
- n_{s_j} is the number of diseases in which the symptom s_j appears in the whole relationship.

The module then calculates the importance of the disease-personal health attribute relationship for each disease in the disease-personal health attribute relation graph G'. To calculate the importance of the disease-personal health attribute relationships, we also exploit the equation that combines the PageRank and modified TF-IDF presented in Eq. (3). The importance of the disease-personal health attribute relations for a specific disease d_m within possible diagnoses D can be calculated as follows.

$$DP(d_m) = (1-d)+d\cdot \sum\nolimits_{(p_i \in In(d_m))} TF(d_m, p_i) \frac{DS(p_i)}{(|Out(p_i)|)}$$
$$TF(d_m, p_i) = \frac{Fq(d_m, p_i)}{(Total(d_m))} \cdot log(\frac{N}{n(p_i)}) \tag{4}$$

- p_i is a personal health attribute that has a relationship with disease d_m

To support the multi-level diagnosis, we calculate the similarity weight between the diseases that are in the possible diagnoses D from and diseases that are in the user's health keywords K. The similarity weight is calculated based on the shared symptoms between two diseases by exploiting the cosine similarity, which is a widely adopted measure in both text mining and the biomedical literature to quantify the similarity between pairs of concepts. The similarity between disease d_m within possible diagnoses D and disease d_k within user's health keywords K is calculated as follows.

$$Dsim(d_m, d_k) = \frac{\sum_{(j=1)}^{n} d_{m,s_j} d_{k,s_j}}{\sqrt{(\sum_{(j=1)}^{n}(d_{m,s_j})^2)}\sqrt{(\sum_{(j=1)}^{n}(d_{k,s_j})^2)}} \tag{5}$$

- d_{n,s_j} and d_{k,s_j} are the co-occurrence ID frequencies of shared symptom s_j between diseases d_m and d_k.

The module calculates the ranking score for each disease of possible diagnoses D. This ranking score is calculated by considering the importance of the disease-symptom relationships DS (d_m), the importance of the disease-personal health attribute relationships DP (d_m), and the similarity weight between diseases $Dsim$ $(d_m,\ d_k)$. The ranking score of each disease RS (d_m) is calculated as follows.

$$RS(d_m) = \alpha \cdot DS(d_m) + \beta \cdot DP(d_m) + \gamma \cdot \frac{\sum_{i=1}^{k} Dsim(d_m, d_k)}{|D|} \tag{6}$$

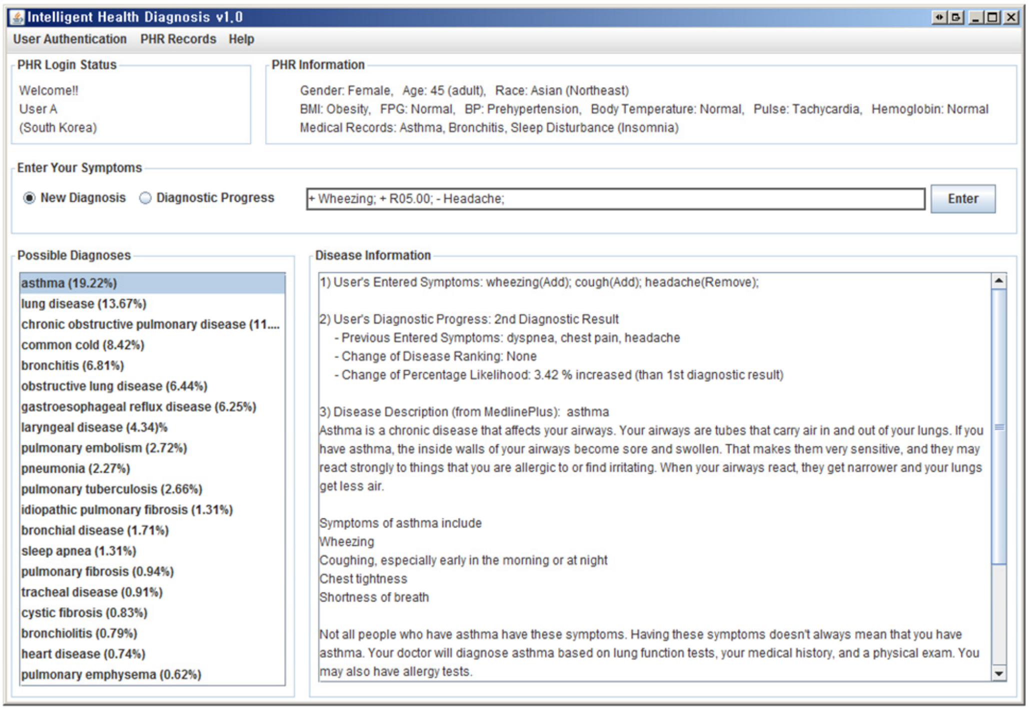
- $\sum_{j=1}^{K} Dsim\left(d_m, d_j\right)$ is the sum of all similarity weights between disease d_m and the diseases within health keywords K
- $|D|$ is total number of diseases in health keywords K

The parameters α, β and $\gamma(0 < \alpha, \beta, \gamma \leq 1)$ are used to balance the contributions when computing the ranking score for each disease. The proposed technique sets the values of $\alpha, \beta,\ \gamma$ as 0.3, 0.5 and 0.2, respectively. (where $\alpha + \beta + \gamma = 1$). After calculating the ranking score for each disease, the module rearranges the possible diagnoses in descending order of the ranking score and selects the top-m diseases. The module then calculates the percentage likelihood of each disease based on the sum of the ranking scores for selected top-m diseases, and provides the ranked list of possible diagnoses, along with their percentage likelihood, as its output.

Finally, to provide the user's diagnostic progress information, the module stores all relation graphs used in the diagnosis in the HDDO as the user's diagnostic result log. The diagnostic result logs are stored in HDDO's *Diagnostic Logs* subclass, and different versions are automatically generated whenever diagnostic results are updated with additional user queries. If the user wants to know the progress of their diagnosis, we recalculate the ranking score via the updated relationships based on the symptoms added or removed by the user. Therefore, we can provide an updated diagnostic result with a change in the ranking or percentage rate of each diagnosis, and this result can be exploited to track or monitor previously diagnosed diseases or other possible diseases.

V. PROTOTYPE IMPLEMENTATION

We implemented a prototype of the proposed technique using Java 1.8, TrOWL version 1.5, and the Jena Ontology API.



PHR Login Status

Welcome!!
User A
(South Korea)

PHR Information

Gender: Female, Age: 45 (adult), Race: Asian (Northeast)
BMI: Obesity, FPG: Normal, BP: Prehypertension, Body Temperature: Normal, Pulse: Tachycardia, Hemoglobin: Normal
Medical Records: Asthma, Bronchitis, Sleep Disturbance (Insomnia)

Enter Your Symptoms

● New Diagnosis ○ Diagnostic Progress

+ Wheezing; + R05.00; - Headache;

Possible Diagnoses

asthma (19.22%)
lung disease (13.67%)
chronic obstructive pulmonary disease (11....
common cold (8.42%)
bronchitis (6.81%)
obstructive lung disease (6.44%)
gastroesophageal reflux disease (6.25%)
laryngeal disease (4.34)%
pulmonary embolism (2.72%)
pneumonia (2.27%)
pulmonary tuberculosis (2.66%)
idiopathic pulmonary fibrosis (1.31%)
bronchial disease (1.71%)
sleep apnea (1.31%)
pulmonary fibrosis (0.94%)
tracheal disease (0.91%)
cystic fibrosis (0.83%)
bronchiolitis (0.79%)
heart disease (0.74%)
pulmonary emphysema (0.62%)

Disease Information

1) User's Entered Symptoms: wheezing(Add); cough(Add); headache(Remove);

2) User's Diagnostic Progress: 2nd Diagnostic Result
- Previous Entered Symptoms: dyspnea, chest pain, headache
- Change of Disease Ranking: None
- Change of Percentage Likelihood: 3.42 % increased (than 1st diagnostic result)

3) Disease Description (from MedlinePlus): asthma
Asthma is a chronic disease that affects your airways. Your airways are tubes that carry air in and out of your lungs. If you have asthma, the inside walls of your airways become sore and swollen. That makes them very sensitive, and they may react strongly to things that you are allergic to or find irritating. When your airways react, they get narrower and your lungs get less air.

Symptoms of asthma include
Wheezing
Coughing, especially early in the morning or at night
Chest tightness
Shortness of breath

Not all people who have asthma have these symptoms. Having these symptoms doesn't always mean that you have asthma. Your doctor will diagnose asthma based on lung function tests, your medical history, and a physical exam. You may also have allergy tests.

FIGURE 12. **Prototype interface.**

We also used Microsoft's HealthVault Java SDK[6] to synchronize our prototype with a PHR service. The HealthVault is a well-known web-based PHR service for storing and maintaining health information. Fig. 12 shows the prototype of our proposed technique.

First, when a user logs into the prototype system, their demographic information and analyzed health conditions are displayed in synchronization with the PHR service. The user can then query the symptoms by selecting whether to execute a new diagnosis or to execute the diagnostic progress based on previous diagnostic results. After obtaining the possible diagnoses from the HDDO, the prototype system provides a ranked list of possible diagnoses with their percentage likelihoods next to each possible disease. Finally, the prototype system provides information about each diagnosed disease through the disease information text field. In particular, it provides information about the user's entered symptoms can be displayed in various forms, such as synonyms, abbreviations, and codes, and information about

whether symptoms have been added or removed is provided. Additionally, if the user selects the diagnostic progress when querying for symptoms, the prototype system provides information on changes in the disease rank or the percentage likelihood based on previous diagnostic results. To deliver trustworthy information about possible diseases, we provide information for each disease acquired from MedlinePlus, which is a web-based health information resource. This information contains a description of certain diseases, including definitions of the disease, symptoms, diagnosis procedures, and treatments. Thus, from this information, user can determine whether the analyzed PHR data are sufficient for diagnosis or whether other specific diagnostic tests are required.

VI. EXPERIMENTAL RESULTS

In this section, we provide a detailed description of the performance evaluation of our intelligent health diagnosis technique. The performance of our proposed technique was evaluated through four experiments: 1) optimization of the parameters α, β, and γ, 2) evaluation of diagnostic accuracy through comparison with existing well-known symptom checkers and other related approaches, 3) evaluation of bias in terms of diagnostic accuracy depending on the type of diseases, and 4) effectiveness of terminological knowledge. We first introduce the datasets and evaluation metrics for evaluating our technique and then discuss the results of the performance evaluations.

A. DATA SETS

Obtaining actual patient data is extremely difficult because of issues such as privacy regulations [33]. Thus, we present indirect evaluations of how well the proposed technique would perform using simulated datasets. To evaluate the proposed technique, we used combined datasets comprising 45 standardized patient vignettes presented by Semigran *et al.* [7] and medical diagnosis lists from the medical book "*Current Essentials of Medicine* " [34]. The 45 standardized patient vignettes were used as the gold standard in evaluating the

[6] https://opensource.microsoft.com/?tag=healthvault

accuracy of the diagnoses provided by 23 existing online symptom checkers. However, we determined that the number of patient vignettes was not sufficient to demonstrate the generality and validity of the proposed technique. Therefore, we added 297 medical diagnosis lists from [34] to extend the dataset to a total of 342 diagnostic cases. Both datasets present diagnosis cases for the patients' diseases, consisting of a set of symptoms, demographic information, and medical records; they were created by various clinical sources, including materials used to educate health professionals, and were reviewed by several medical experts. Additionally, both datasets reflected various ranges of diagnoses, including common to less common and from low acuity to life-threatening diseases. Therefore, they have been widely used as a gold standard for evaluating medical diagnosis systems [5], [35]. We further categorized the diagnostic cases in the combined dataset into five disease groups to evaluate the bias in terms of diagnostic accuracy according to the type of diseases. A detailed explanation of why we categorized the five disease groups is given in Section VI.E. Table 2 presents the categorized disease groups and the number of diagnostic cases used in the performance evaluations. Finally, for the performance evaluations, we entered the patient information for each diagnostic case into the PHR service and synchronized it with our proposed technique. Based on the survey by Riches *et al.* [36], the size of our dataset is sufficiently large to evaluate the performance compared to other academic research work.

B. EVALUATION METRIC

We used the Mean Reciprocal Rank to the k-th position ($MRR@k$) as the performance criterion to evaluate the diagnostic accuracy. This is a commonly used information retrieval metric for measuring the accuracy of ranked retrieval results and is well-suited to systems in which only the first result matters [37]. $MRR@k$ refers to the mean of the reciprocal rank (RR) for each the top-k result. $RR@k$ refers to the multiplicative inverse of the rank of the first relevant item in the top-k results returned by our proposed technique. If there is no relevant item is contained in the top-k results, a value of zero is returned. For given a set of queries Q, $MRR@k$ can be formally defined as

$$MRR@k = \frac{1}{|Q|}\sum\nolimits_{i=1}^{|Q|}\frac{1}{Rank_i} \tag{7}$$

where $|Q|$ is the number of queries and $Rank_i$ denotes the rank position of the first relevant disease for the i-th query in the top-k results.

In our experiments, |Q| denotes the dataset size (i.e., 342 diagnostic cases). In addition, the value of k was set to 1, 3, or 20 (i.e., *MRR@1*, *MRR@3*, or *MRR@20*) because it is important to measure changes in diagnostic accuracy when the diagnosed disease is within the top 1, 3, or 20 results. In the case of results outside the top 20, we determined that the list of diagnostic results would have been overly long and unlikely to be useful for users.

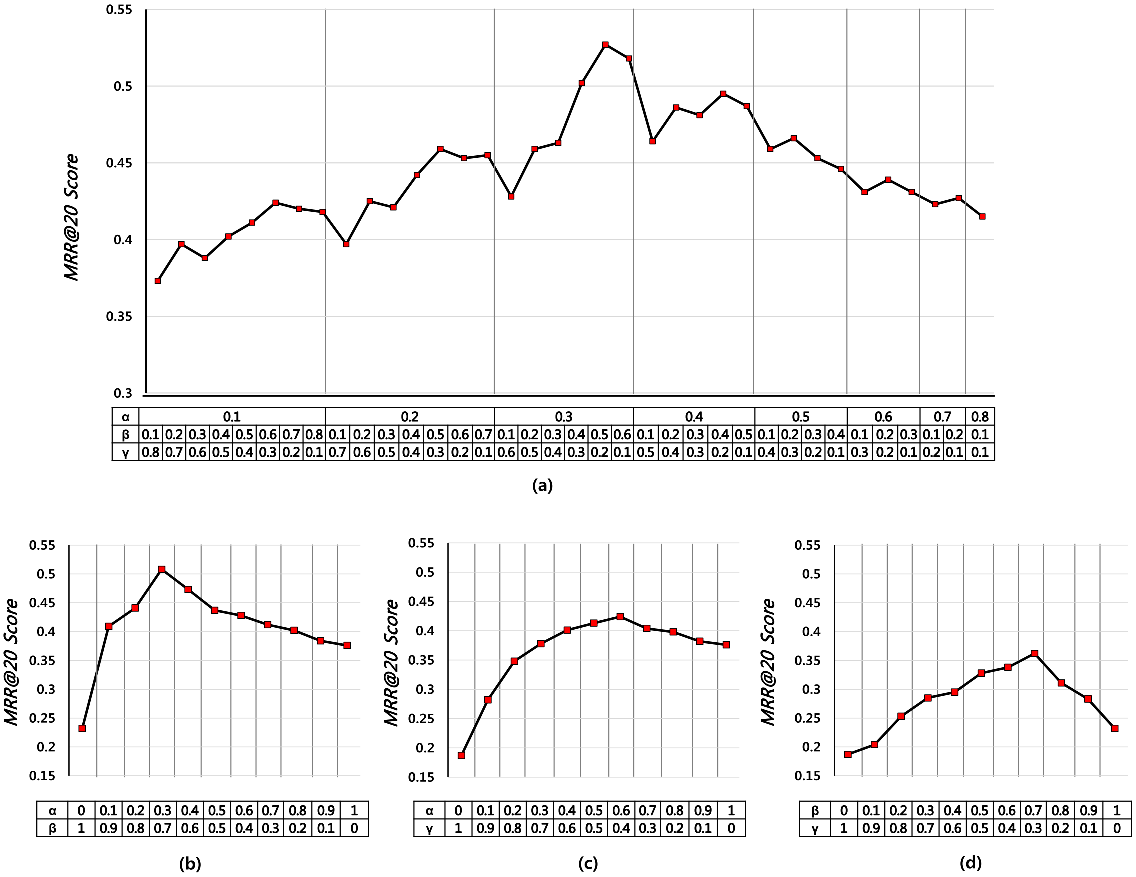
C. PARAMETER OPTIMIZATION

The value of parameter α, β, and γ used in Eq. (6) to calculate the ranking score of each disease can affect the accuracy of our diagnosis results. The experiment thoroughly analyzed the impact of parameter values on accuracy using the *MRR@20* score. To determine the optimal parameter values, we varied the values of parameter α, β, and γ in Eq. (6) from 0 to 1. Fig. 13(a) illustrates the optimized values of parameter α, β, and γ in our proposed technique. The highest *MRR@20* score indicates the optimal values of the parameters α, β, and γ.

As shown in Fig. 13(a), the optimal performance (i.e., highest *MRR@20* score) was achieved when the parameters α, β, and γ were set to 0.3, 0.5 and 0.2, respectively. Therefore, we adopted the optimal value of parameters α, β, and γ which were empirically adopted as settings required to provide diagnostic results to users. However, it was still necessary to investigate why the diagnostic accuracy changes when α, β, and γ are assigned different values. To investigate this, we considered only the relationships between sets of two parameters in Eq. (6) and analyzed how diagnostic accuracy was affected when the values of each parameter were changed.

In Fig. 13(b), *WithoutDsim* shows the results of diagnostic accuracy when only parameters α and β (e.g. $\alpha + \beta = 1$, $\gamma = 0$) are considered. In *WithoutDsim*, increasing the value of parameter α means that the contribution of the importance of the disease-symptom relationships increases in the ranking score for each disease. Contrarily, increasing the value of parameter β means that the contribution of the importance of the disease-personal health attribute relationships increases in the ranking score for each disease. From the results of *WithoutDsim*, we found that optimal performance is achieved when α and β are set to 0.3 and 0.7, respectively. This demonstrates that the combination of α and β showed better performance than when only one of these parameters were considered, and both the importance of the disease-symptom relationships and the importance of the disease-personal health attribute relationships are significant in diagnosing diseases. Further, since the value of β was higher than that of α in the optimal performance of *WithoutDsim*, the importance of the disease-personal health attribute relationships is more significant than the importance of the disease-symptom relationships to the proposed technique. Thus, we confirmed that the parameter β should be set higher than α to obtain optimal performance.

In Fig. 13(c), *WithoutDP* shows the diagnostic accuracy when only the parameters α and γ (e.g. $\alpha + \gamma = 1, \beta = 0$) are considered. Similarly, in Fig. 13(d), *WithoutDS* shows the diagnostic accuracy when only the parameters β and γ (e.g. $\beta + \gamma = 1$, $\alpha = 0$) were considered. The *WithoutDP* and *WithoutDS* results show, respectively, that optimal performance is achieved when α and γ are set to 0.6 and 0.4, respectively, and when β and γ are set to 0.7 and 0.3, respectively. Therefore, the combination of parameters α and γ in *WithoutDP* showed better performance than when



(a)

(b)

(c)

(d)

FIGURE 13. Parameter optimizations.

only one of these parameters was considered. Likewise, in *WithoutDS*, the combination of parameter β and γ showed better performance than when only one of these parameters was considered. In addition, the value of parameter γ was generally lower than parameters α or β in the optimal performance of *WithoutDP* and *WithoutDS*. Thus, we identified that the similarity weight between the diseases is less significant than that of the disease-symptom relationships and disease-personal health attribute relationships in the proposed technique. Consequently, we can confirm that the value of parameter γ should be set lower than the values of parameters $\alpha$and β to obtain the optimal performance.

D. EVALUATION OF DIAGNOSTIC ACCURACY

To evaluate the diagnostic accuracy of the proposed technique, we carried out two comparative experiments: 1) comparison with existing symptom checkers, and 2) comparison with related approaches.

1) COMPARISON WITH SYMPTOM CHECKERS

To evaluate the accuracy of the proposed technique, we compared it with six other symptom checkers by computing *MRR@1*, *MRR@3*, and *MRR@20*. We first selected five currently available symptom checkers: *Isabel*,[7] *MayoClinic*,[8] *Sympcat*,[9] *AskMD*,[10] and *WebMD*.[11] These symptom checkers demonstrated generally superior diagnostic accuracy among the 23 checkers evaluated by Semigran *et al.* [7]. We also selected *WithoutPHR* to investigate the accuracy when individual health such as disease-personal health attribute relationships and similarity between diseases were not considered. *WithoutPHR* shows the results of accuracy when we only considered the importance of the disease-symptom relationships in Eq. (6) (i.e., $\alpha = 1$ and β, $\gamma = 0$). Fig. 14 compares the diagnostic accuracy of the proposed technique with those of the other six symptom checkers.

The results in Fig. 14 show that the proposed technique gives the highest score in all *MRRs*, which means that the proposed technique provides more accurate diagnostic results with than all the other comparative symptom checkers. Hence, the strategies of the proposed technique that exploit personal health attributes obtained from a PHR service and

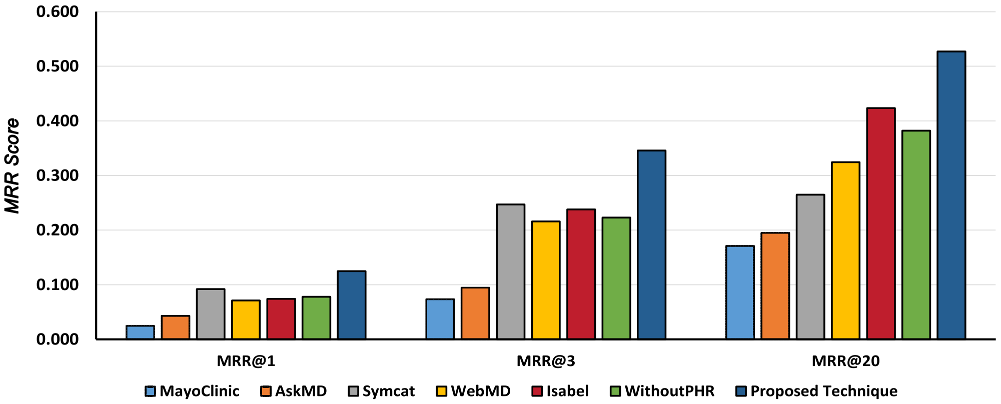
[7] https://symptomchecker.isabelhealthcare.com/

[8] https://www.mayoclinic.org/symptom-checker/select-symptom/itt-20009075

[9] http://www.symcat.com/

[10] https://www.sharecare.com/askmd/get-started

[11] https://symptoms.webmd.com/default.htm



	MayoClinic	AskMD	Symcat	WebMD	Isabel	WithoutPHR	Proposed Technique
MRR@1	0.024	0.043	0.092	0.071	0.074	0.078	0.125
MRR@3	0.074	0.095	0.231	0.199	0.213	0.223	0.346
MRR@20	0.171	0.195	0.265	0.324	0.423	0.382	0.527
Average MRR	0.090	0.111	0.201	0.204	0.245	0.228	0.333

FIGURE 14. Comparison of the diagnostic accuracy with existing symptom checkers.

possible diagnoses derived from our automatically generated HDDO appear to be more effective than those of other existing symptom checkers.

Among the existing symptom checkers, *Isabel* and *WebMD* achieved better accuracy than the others. These two-symptom checkers ask for the user's demographic information, such as age and gender, to obtain diagnostic results, and provide terminological processes for handling syntactic variants of user inputs such as plural forms. As with our technique, *Isabel* allows users who know their health condition and diseases to manually enter them together with their symptoms. For this reason, *Isabel* outperformed *WithoutPHR* by 6.7% (*MRR@3*) and 10.7% (*MRR@20*), as the latter only considers the importance of the disease-symptom relationships. However, *Isabel* could not recognize some of the synonym terms in the user inputs and did not support the terminology process for abbreviations and codes. Thus, the accuracy of *Isabel* was 5.1% lower than that of *WithoutPHR* in terms of *MRR@1*. In addition, since *Isabel* ignores unrecognized symptoms and the user's health attributes, the *MRR@1*, *MRR@3*, and *MRR@20* scores were 40.8%, 31.2%, and 19.7% lower than those of the proposed technique, respectively. The *MRR@1, MRR@3,* and *MRR@20* scores of *WebMD* were 9%, 3.1%, and 15.1% lower than that of *WithoutPHR* and 43.2%, 37.6%, and 38.4% lower than that of the proposed technique, respectively. This is because *WebMD* is unable to enter the user's health conditions and currently suffering diseases as user inputs. Additionally, as all MRR scores achieved by *WebMD* were lower than those of *WithoutPHR,* which does not consider the user's demographic information, we found that *WebMD* does not effectively incorporate demographic information into its diagnosis process. Thus, in the evaluation, *WebMD* predominately provided general diseases based on simple pairwise relationships with the user-entered symptoms.

The next most-accurate symptom checker following *WebMD* was *Symcat*. Unlike other symptom checkers, *Symcat* uses real patient data to inform users of the diseases suffered by other users who experienced the same symptoms. However, *Symcat* provides only six listed diagnoses as results to users. Semigran *et al.* [7] reported that there was no significant difference in evaluating the correct diagnosis in the top 20 between symptom checkers that listed more than 20 diagnoses compared with those that only listed 1-3 diagnoses. However, their results were obtained by testing only 45 patient cases, and *Symcat* showed a significant difference in *MRR@1*, *MRR@3*, and *MRR@20* for the additional diseases and diverse patient cases included in our dataset. We found that the real patient data used by *Symcat* helps to increase the diagnostic accuracy, because *Symcat* was more accurate in terms of *MRR@1* and *MRR@3* than the other existing symptom checkers. However, *Symcat* is less accurate in terms of *MRR@20* because it provides a limited number of diagnostic results compared with the other symptom checkers. In addition, since *Symcat* manually assigns patient data to their knowledge model, it is unable to provide recently issued diseases such as the Zika virus. Correspondingly, the *MRR@1, MRR@3,* and *MRR@20* scores of *Symcat* were 26.4%, 28.6%, and 49.8% lower than those of the proposed technique.

Finally, *AskMD* and *MayoClinic* delivered relatively lower accuracy than the other symptom checkers. These symptom checkers can only test a few common diseases and allow users to enter only one representative symptom instead of a series of symptoms. To provide diagnostic results to users, *AskMD* requires users to provide demographic information and some health history, whereas *MayoClinic* allows users to enter only symptoms. However, even in this case, *MayoClinic* only permits one symptom to be chosen from a pre-defined list of 46 symptoms. Thus, *AskMD* was more accurate than *MayoClinic* but both symptom checkers delivered mostly inaccurate diagnostic results for our dataset. Consequently, the *MRR@1, MRR@3,* and *MRR@20* scores of *AskMD* were 65.6%, 72.6%, and 63% lower than those of the proposed technique, whereas *MayoClinic* scored 80.5%, 78.7%, and 67.5% lower, respectively.

2) COMPARISION WITH RELATED APPROACHES

To compare the accuracy of the proposed technique with those of related approaches, we conducted an experiment using four knowledge model construction approaches: 1) *TEXT*, a health diagnosis based on the relationships between diseases and symptoms found in online health documents presented by Mohammed *et al.* [20]; 2) *META*, a health diagnosis based on the relationships between diseases and symptoms found using MetaMap presented by Okumura *et al.* [21]; 3) *PubM*, a health diagnosis based on the relationships between diseases and symptoms found using PubMed article IDs and MeSH metadata presented by Zhou *et al.* [22]; 4) *MulD*, a health diagnosis based on multi-level diagnosis presented by Rodriguez-Gonzalez and Alor-Hernandez [10] and Rodriguez-Gonzalez *et al.* [27] . To compare diagnostic accuracy when individual health is not considered, we used *WithoutPHR*. Conversely, to compare diagnostic accuracy when multi-level diagnosis was considered, we used *WithoutDP*, which was discussed in Section VI.C.

The following preprocessing procedures and settings were applied in the experiments. For *TEXT* and *META*, which require online health documents in advance, we collected documents related to diseases and symptoms from 13 well-known online health websites including the National Health Service (NHS),[12] MedlinePlus,[13] and MedicineNet.[14] In the case of *MulD*, as the knowledge model was created manually, we constructed a model using the relationships between diseases and symptoms acquired from PubMed and applied their multi-level diagnosis algorithms. After finding the relationships between diseases and symptoms in four related approaches, we applied our expanded terminological knowledge for disease and symptom terms to these in order to establish a fair experimental environment for the lexical variants of the user inputs. Finally, to calculate the ranking score for each disease, *PubM* used TF-IDF and the other models used Eq. (6). Fig. 15 compares the accuracy of our proposed technique with those of the related approaches.

As shown in Fig. 15, the proposed technique outperformed the other approaches. The second-best was *WithoutDP*. This result indicates that our strategies for multi-level diagnosis (i.e., similarity weight between diseases) are likely to improve the diagnostic accuracy, although personal health attributes were not considered in the diagnostic accuracy. Hence, in terms of the diagnostic accuracy, we can confirm that consideration of the disease-symptom relationships, disease-personal health attribute relationship, and the similarities between diseases are all significant factors in the knowledge model construction and disease ranking calculation.

WithoutPHR and *PubM* were more accurate than *TEXT* and *META* in terms of all *MRRs*. This indicates that the relationships found using the PubMed article IDs and MeSH metadata are more accurate than those found in online health documents through text analysis. However, the *MRR@1, MRR@3,* and*MRR@20* scores of *PubM* were 9%, 8.5%, and 13.4% lower than those of *WithoutPHR* and 43.2%, 41%, and 37.2% lower than those of the proposed technique, respectively. This means that our strategy of combining modified TF-IDF and PageRank algorithms is more accurate than *PubM* 's strategy of using TF-IDF alone. This is because we consider not only the co-occurrence of IDs between diseases and symptoms, but also the link information of diseases and symptoms.

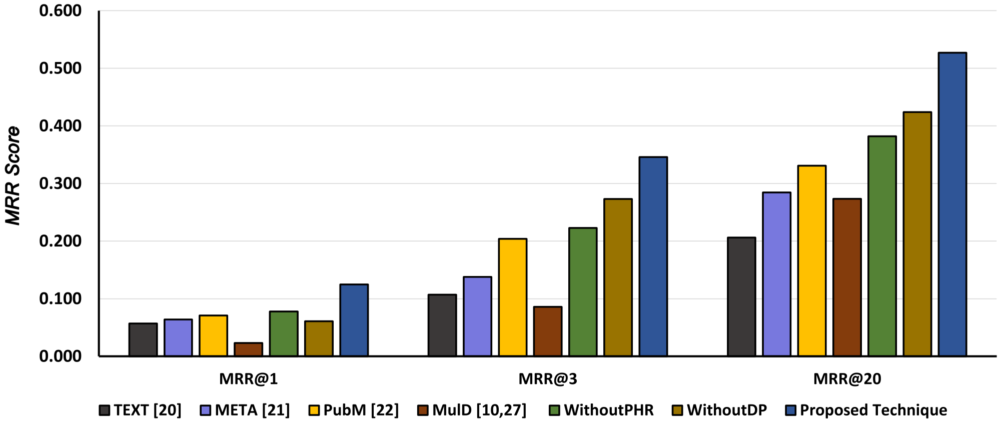
The next most-accurate approach was *META*, followed by *MulD*. *TEXT* delivered the lowest accuracy compared to all the other approaches. The result that *META* was more accurate than *TEXT* means that supporting lexical variants for diseases and symptom terms is an important factor in finding the relationships between diseases and symptoms. However, *META* could not find relationships using abbreviations or codes for disease or symptom terms and could not filter out relationships obtained from negative mentions. Furthermore, *META* ignored disease and symptom terms in online health documents because it often received incorrect concepts through MetaMap. For example, the word "cold" which means "common cold" was sometimes ignored because it was assigned as "*cold temperature* (*natural phenomenon*)" or "*cold sensation* (*physiologic function*)" by MetaMap. As a result, the *MRR@1, MRR@3,* and*MRR@20* scores of *META* were 17.9%, 38.1%, and 25.5% lower than those of *WithoutPHR* and 48.8%, 60.1%, and 46% lower than those of the proposed technique, respectively.

Finally, although *MulD* is based on multi-level diagnosis that considers the user's current diseases as diagnostic elements, it was much less accurate than *WithoutDP*, and even worse than *WithoutPHR*, *PubM*, and *META*. To represent the concepts of multi-level diagnosis, the algorithms of *MulD* assume that a disease can constitute a symptom of another disease. This means that, if disease X has symptoms *SymA*, *SymB*, and disease Y, and disease Y has symptoms *SymC* and *SymD*, then disease X will eventually have symptoms *SymA*, *SymB*, *SymC*, and *SymD*. Consequently, to obtain diagnostic results, all the symptoms of the user's current diseases are used in conjunction with the user's entered symptoms. However, *MulD* does not look at demographic information in the diagnosis process. In addition, we found that *MulD* was defective in performing normal diagnoses that do not require the input of the user's current diseases as diagnostic elements. As the *MulD* algorithms are solely used for multi-level diagnosis, the symptoms of diseases that

[12] https://www.nhs.uk/pages/home.aspx

[13] https://medlineplus.gov/

[14] https://www.medicinenet.com/script/main/hp.asp



	Text [20]	META [21]	PubM [22]	MulD [10,27]	WithoutPHR	WithoutDP	Proposed Technique
MRR@1	0.057	0.064	0.071	0.023	0.078	0.061	0.125
MRR@3	0.107	0.138	0.204	0.086	0.223	0.273	0.346
MRR@20	0.206	0.284	0.331	0.273	0.382	0.424	0.527
Average MRR	0.123	0.162	0.202	0.127	0.228	0.253	0.333

FIGURE 15. Comparison of the diagnostic accuracy with related approaches.

do not actually occur in the user are used for diagnosis. Furthermore, although the user's diseases are linked to other diseases, it is not clear whether the symptoms of the user's disease are associated with other diseases as well. As a result, the *MRR@1, MRR@3,* and*MRR@20* scores of *MulD* were 62.3%, 68.5%, and 35.5% lower than those of *WithoutDP*, and 70.5%, 61.4%, and 28.5% lower than those of *WithoutPHR*, respectively. The *MRR@1*, *MRR@3*, and *MRR@20* scores of *MulD* were 81.6%, 75.1%, and 48.1% lower than those of the proposed technique, respectively.

E. EVALUATION OF BIAS IN TERMS OF DIAGNOSTIC ACCURACY DEPENDING ON THE TYPE OF DISEASE

Diagnostic accuracy should be ensured for various types of diseases, not only for certain types of diseases. Therefore, evaluation of bias in diagnostic accuracy is important in ensuring the performance of our proposed technique. To evaluate diagnostic accuracy bias, we categorized our dataset into the five disease groups listed in Table 2. These categorizations were made for the following reasons.

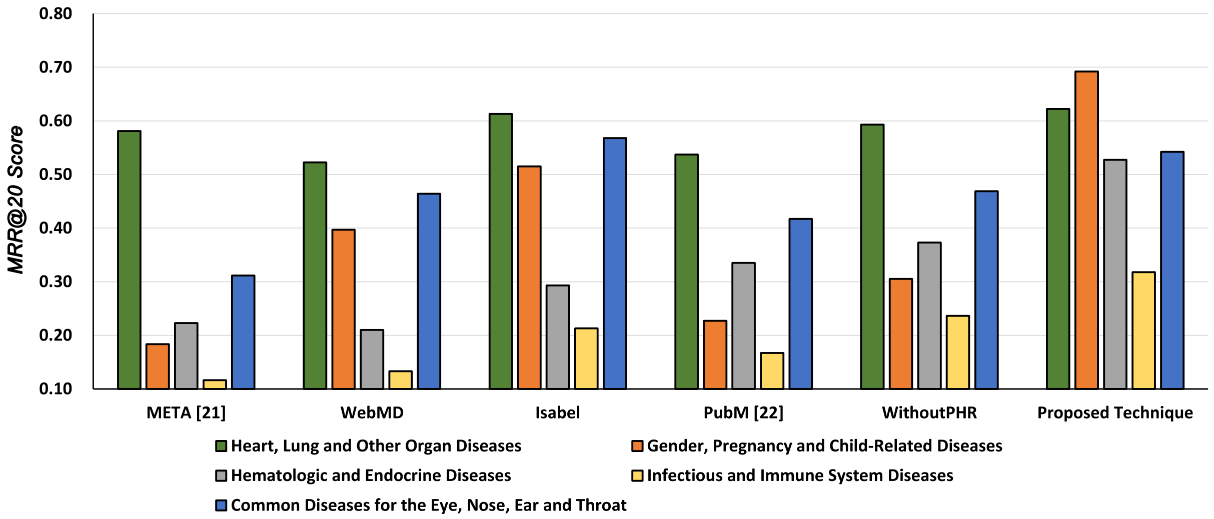
First, diseases related to the heart, lung, and other organs can be diagnosed from relatively obvious symptoms compared to other diseases. Thus, to investigate whether accuracy is biased toward user-entered symptoms, we categorized related diseases into a "*heart, lung, and other organ diseases* " group. In the "*gender, pregnancy, and child-related diseases* " group, we categorized the related diseases to investigate whether accuracy is biased by user demographic information. To investigate whether accuracy is biased by user health conditions, we created two groups: "*hematologic and endocrine diseases*" and "*infectious and immune system diseases* ". In the "*hematologic and endocrine diseases* " group, we categorized diseases that primarily require pre-screening through well-known health measures, whereas in the "*infectious and immune system diseases* " group, we categorized diseases that require the user's current disease information or medical histories. Finally, we categorized common diseases that often appear in users under a "*common diseases of the eye, nose, ear, and throat* " group.

To measure bias, we figured out the *MRR@20* score of each disease group in the diagnostic results and calculated the coverage score for types of disease that exploit standard deviation to measure whether the respective disease groups are well-distributed in the diagnostic results. In this case, a low standard deviation value indicates that diagnostic accuracy is evenly distributed across various types of disease. Therefore, the coverage score for types of disease is the inverse of the standard deviation for disease group distribution. The coverage score can be calculated as follows:

$$Cov\,(System) = 1 \Bigg/ \sqrt{\frac{\sum_{i=1}^{n} (x_i - \bar{x})^2}{M}} \tag{8}$$

where x_i is the *MRR@20* score for each disease group, $\bar{x}$ is the mean of x_i, and $M$is the number of disease groups, which in this case is five.

To perform comparison experiments, we used the five systems described in the previous section. For comparison with existing symptom checkers, we selected *Isabel* and*WebMD,* which provided to be more accurate than other symptom checkers. For comparison with related approaches for automatic knowledge model construction, we selected *PubM,* which uses relationships found from PubMed, and *META,* which uses relationships found from online health documents. Finally, we compared *WithoutPHR* to investigate the



	META [21]	WebMD	Isabel	PubM [22]	WithoutPHR	Proposed Technique
Heart, Lung, and Other Organ Diseases	0.58	0.56	0.61	0.56	0.59	0.62
Gender, Pregnancy, and Child-Related Diseases	0.18	0.29	0.52	0.25	0.31	0.69
Hematologic and Endocrine Diseases	0.22	0.24	0.32	0.29	0.34	0.53
Infectious and Immune System Diseases	0.12	0.14	0.21	0.17	0.24	0.32
Common Diseases for the Eye, Nose, Ear, and Throat	0.31	0.47	0.57	0.44	0.47	0.55
Standard Deviation	0.163	0.155	0.153	0.139	0.127	0.126
Coverage	6.15	6.47	6.54	7.17	7.89	7.93

FIGURE 16. **Performance comparison of the effect of bias on diagnostic accuracy.**

bias toward accuracy when individual health is not considered. Fig. 16 shows performance comparisons of the effect of bias on diagnostic accuracy.

As shown in Fig. 16, the proposed technique performed better than other systems in terms of the coverage score. This means that the diagnostic accuracy of the proposed technique is less biased than the other comparison systems. Therefore, we can clearly confirm that the proposed technique has superior performance in ensuring diagnostic accuracy for various types of diseases. In particular, in the comparison with *WithoutPHR*, we found that considering the individual's health in the diagnosis process has significant effects in terms of reducing bias by providing more accurate diagnoses for various types of diseases. The proposed technique produced worst results for the "*infection and immune system diseases* " group compared to the other disease groups, although the other systems also produced their worst results for this group. However, the overall results produced by the proposed method for this group were still better than those produced by the other systems. Thus, we confirmed that the strategies of the proposed technique are not wrong. The diseases in the "*infection and immune system diseases*" group are typical viral diseases such as cholera, malaria, and measles. These types of diseases can occur as a result of a variety of factors, including ingestion of rotten foods or drinks, inhalation of pathogenic agents, and contact with others. These factors are related to the lifestyle of the user rather than their health record and, because lifestyle varies from person to person, it is difficult to represent this group as patterned. It is therefore, to date, not properly recorded in PHR services. For this reason, the proposed technique, which uses information on personal health recorded in PHR services, had the lowest accuracy in the "*infection and immune system diseases*" group along with the other approaches.

From the coverage scores, the next less biased system was *PubM,* followed by *Isabel* and *WebMD,* which are existing symptom checkers. Thus, we have identified that automatic knowledge model construction methods based on PubMed's article ID and MeSH metadata can provide less bias in diagnostic accuracy than the manually constructed knowledge models. In a comparison with *WithoutPHR*, we found that the TF-IDF of *PubM* which was used to rationalize the diagnostic accuracy, is not helpful to reduce the bias. Furthermore, as PubM does not take a user's individual health into account in its diagnostic process, its diagnostic accuracy was biased toward the "*heart, lung, and other organ diseases*" and "*common diseases of the eye, nose, ear, and throat*" groups.

Isabel considers user demographic information in the diagnostic processes. Thus, *Isabel* was more accurate than *WithoutPHR* for the "*gender, pregnancy and* *child-related diseases"* group. However, for the "*hematologic and endocrine diseases"* and "*infectious and immune system disease"* groups, which require consideration of the user's health condition, *Isabel* was less accurate than *WithoutPHR*. *WebMD* also considers user demographic information in the diagnostic processes, but it was lower than *WithoutPHR* for the "*gender, pregnancy and child-related diseases* " group. This signifies that *WebMD* does not effectively incorporate demographic information into its diagnosis process. From this we can conclude that the knowledge models used by these two-symptom checkers do not effectively consider the user's health condition or current disease information in their diagnostic processes.

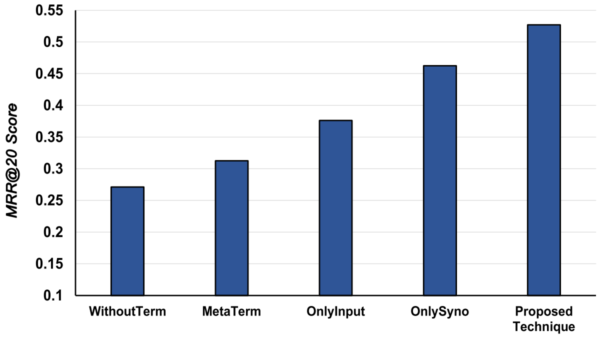
Finally, *META* showed the most bias on accuracy results compared to all other systems. As most of the diagnostic accuracy for *META* corresponded to "*heart, lung, and other organ diseases"* group, it delivered the lowest coverage score over the various disease group types. Thus, we can confirm that MetaMap, which supports lexical variance in online health documents, can be helpful in diagnosing diseases with relatively obvious symptoms, but it is not very helpful for other types of diseases, especially those related to user health.

F. EFFECTIVENESS OF TERMINOLOGICAL KNOWLEDGE

The proposed technique exploits terminological knowledge acquired from BioPortal to support lexical variants in user inputs and find article IDs related to disease and symptom terms from PubMed. Correspondingly, it is necessary to demonstrate how terminological knowledge affects diagnostic accuracy. To investigate this, we performed comparisons involving the following four cases: 1) *WithoutTerm*, a case that did not use terminological knowledge; 2) *MetaTerm*, a case that exploited MetaMap to acquire terminological knowledge; 3) *OnlyInput*, a case that used terminological knowledge only to support lexical variants of user inputs; and 4) *OnlySyno*, a case that used terminological knowledge that acquired only synonyms of disease and symptom terms from BioPortal. In this evaluation, we used the same datasets as in the evaluation of diagnostic accuracy. Fig. 17 shows accuracy comparisons of the four cases in terms of terminological knowledge.

As depicted in Fig. 17, *WithoutTerm* showed the worst diagnostic accuracy in all cases, especially with 48.6% lower accuracy than the proposed technique. As we expected, it is not appropriate to diagnose a disease without considering its related terminological knowledge; *WithoutTerm* ignores unrecognized symptoms or personal health attributes provided by users. This result demonstrates the necessity of applying a strategy to overcome issues derived from the peculiarities of medical terminology for performing health diagnoses.

MetaTerm shows 16.9%, 32.4%, and 40.7% lower accuracy than *OnlyInput*, *OnlySyno*, and the proposed technique, which utilized terminology knowledge acquired from BioPortal. This is because terminological knowledge acquired through MetaMap did not properly capture the abbreviations or codes of the disease or symptom terms. In addition, *MetaTerm* was unable to specify some relationships between diseases and symptoms because it could not find the PubMed article ID associated with the respective terms due to incorrectly assigned UMLS concepts.



	WithoutTerm	MetaTerm	OnlyInput	OnlySyno	Proposed Technique
MRR@20	0.271	0.313	0.376	0.462	0.527

FIGURE 17. Accuracy comparisons from terminological knowledge experiments.

OnlyInput and *OnlySnyo* show 28.6% and 12.3% lower accuracy than the proposed technique, and the result of the proposed technique shows the best accuracy performance compared to all other cases. Compared to *OnlyInput*, we found that it is more effective to consider the terminological knowledge acquired from BioPortal when specifying the relationships from PubMed. Disease and symptom terms extracted from DO and SYMP were generally not exact matches with the terms in PubMed due to the peculiarities of medical terminology. For example, the term "*type 2 diabetes mellitus"* in DO is defined as "*diabetes mellitus, type 2"* in PubMed, and the term *"alzheimer's disease"* in DO is defined as *"alzheimer disease"* without apostrophe 's. As a result, many extracted disease and symptom terms were filtered out in the process of checking whether a related article exists in PubMed and the corresponding relationships could not be specified. Thus, considering the terminological knowledge acquired from BioPortal in specifying the relationships from PubMed produced more enriched disease and symptom terms that could be contained in the relationships. In addition, the results of the proposed technique were always better than *OnlySyno*. This means that it is not sufficient to consider only the synonym of disease or symptom terms when specifying the relationships from PubMed.

From these experiments, we confirmed that terminology knowledge acquired from BioPortal for synonym, abbreviations, and codes effectively increases diagnostic accuracy, and it is more suitable than using other lexical variant support tools to provide diagnostic results to users.

VII. CONCLUSION

In this paper, we have proposed an intelligent health diagnosis technique that provides reliable diagnostic results to users, thus helping them to take appropriate healthcare actions. Using three types of biomedical resources, the proposed technique automatically generates the HDDO and exploits it as a knowledge model. In addition, even if the user enters only symptom-based queries, the technique identifies possible diagnoses considering the user's personal health by synchronizing with PHR services. Moreover, it delivers ranked results of possible diagnoses to users based on methods that elevate the ranking of diagnoses related to the user's health. After initial diagnosis, the proposed technique provides the user's diagnostic progress information through the diagnostic result log which can be stored in the HDDO. Experimental results demonstrate that the proposed technique provides enhanced diagnostic accuracy over existing symptom checkers and related approaches.

However, if the proposed technique is seen as a replacement for a physician, it is likely an inferior alternative. It is believed that physicians have a diagnostic accuracy rate of 85-90%, but the performance of the proposed technique was much lower, as seen in the experiments. Therefore, the diagnostic results provided by the proposed technique are not intended to replace the diagnosis of medical experts and cannot be used as the definitive diagnosis for medical treatments. A definitive diagnosis must be rendered by medical experts. We only claim that our proposed technique can be used as a complementary healthcare tool that provides more reliable diagnostic guidance than currently existing symptom checkers and related approaches. Also, other limitations of our proposed technique are that we only considered diseases, symptoms, and personal health attributes that are recorded in PHR services, even though a diagnosis might be affected by a wide range of attributes including exercise, nutrition, and lifestyle. Furthermore, although more datasets were used than in other academic studies, a much more intensive evaluation would be needed to further develop this technique into a feasible application.

In future work, we plan to examine many more test cases to evaluate and improve the accuracy of our proposed technique. In addition, we will investigate the relation between user lifestyle and health diagnosis using deep learning approaches that automatically extract and learn the features (i.e., chief complaints including symptoms) that need to be diagnosed from multiple online health documents. Finally, we will attempt to expand the proposed technique by implementing multi-document summarization and video refinement techniques. Through these enhancements, the proposed technique will become capable of providing trustworthy summarizations of related articles and refined video content for each diagnosis using information obtained from a variety of reliable health websites.

REFERENCES

[1] K. Khoo, P. Bolt, F. E. Babl, S. Jury, and R. D. Goldman, "Health information seeking by parents in the Internet age," *J. Paediatrics Child Health*, vol. 44, nos. 7–8, pp. 419–423, Jun. 2008.

[2] W. Jacobs, A. O. Amuta, K. C. Jeon, and C. Alvares, "Health information seeking in the digital age: An analysis of health information seeking behavior among US adults," *Cogent Social Sci.*, vol. 3, no. 1, Mar. 2017, Art. no. 1302785.

[3] R. W. White and E. Horvitz, "Cyberchondria: Studies of the escalation of medical concerns in Web search," *ACM Trans. Inf. Syst.*, vol. 27, no. 4, pp. 1–37, Nov. 2009.

[4] F. North, P. Varkey, B. Laing, S. S. Cha, and S. Tulledge-Scheitel, "Are e-health Web users looking for different symptom information than callers to triage centers?" *Telemed. e-Health*, vol. 17, no. 1, pp. 19–24, Feb. 2011.

[5] H. L. Semigran, D. M. Levine, S. Nundy, and A. Mehrotra, "Comparison of physician and computer diagnostic accuracy," *JAMA Internal Med.*, vol. 176, no. 12, pp. 1860–1861, Dec. 2016.

[6] L. Powley, G. Mcllroy, G. Simons, and K. Raza, "Are online symptoms checkers useful for patients with inflammatory arthritis?" *BMC Musculoskeletal Disorders*, vol. 17, no. 1, p. 362, Aug. 2016.

[7] H. L. Semigran, J. A. Linder, C. Gidengill, and A. Mehrotra, "Evaluation of symptom checkers for self diagnosis and triage: Audit study," *Brit. Med. J.*, vol. 351, pp. 1–9, Jun. 2015.

[8] R. A. Miller and A. Geissbuhler, "Diagnostic decision support system," in *Clinical Decision Support Systems: Theory and Practice*, 2nd ed. Springer, 2007, ch. 5, pp. 99–125.

[9] M. Rotmensch, Y. Halpern, A. Tlimat, S. Horng, and D. Sontag, "Learning a health knowledge graph from electronic medical records," *Sci. Rep.*, vol. 7, Jul. 2017, Art. no. 5994.

[10] A. Rodríguez-González and G. Alor-Hernández, "An approach for solving multi-level diagnosis in high sensitivity medical diagnosis systems through the application of semantic technologies," *Comput. Biol. Med.*, vol. 43, no. 1, pp. 51–62, Jan. 2013.

[11] C. Zou and H. Deng, "Using fuzzy concept lattice for intelligent disease diagnosis," *IEEE Access*, vol. 5, pp. 236–242, Dec. 2016.

[12] A. G. Karegowda, A. S. Manjunath, and M. A. Jayaram, "Application of genetic algorithm optimized neural network connection weights for medical diagnosis of Pima Indians diabetes," *Int. J. Soft Comput.*, vol. 2, no. 2, pp. 15–23, May 2011.

[13] I. Kononenko, "Machine learning for medical diagnosis: History, state of the art and perspective," *Artif. Intell. Med.*, vol. 23, no. 1, pp. 89–109, Aug. 2001.

[14] S. Yang, R. Wei, J. Guo, and L. Xu, "Semantic inference on clinical documents: Combining machine learning algorithms with an inference engine for effective clinical diagnosis and treatment," *IEEE Access*, vol. 5, pp. 3529–3546, Dec. 2016.

[15] L. H. Son and N. T. Thong, "Intuitionistic fuzzy recommender systems: An effective tool for medical diagnosis," *Knowl.-Based Syst.*, vol. 74, pp. 133–150, Jan. 2015.

[16] Á. García-Crespo, A. Rodríguez, M. Mencke, J. M. Gómez-Berbís, and R. Colomo-Palacios, "ODDIN: Ontology-driven differential diagnosis based on logical inference and probabilistic refinements," *Expert Syst. Appl.*, vol. 37, no. 3, pp. 2621–2628, Mar. 2010.

[17] O. Arsene, I. Dumitrache, and I. Mihu, "Expert system for medicine diagnosis using software agents," *Expert Syst. Appl.*, vol. 42, no. 4, pp. 1825–1834, Mar. 2015.

[18] M. Thirugnanam, T. Thirugnanam, and R. Mangayarkarasim, "An ontology based system for predicting disease using SWRL rules," *Int. J. Comput. Sci. Bus. Inform.*, vol. 7, no. 1, pp. 1–15, Nov. 2013.

[19] V. Rawte and B. Roy, "OBESTDD: Ontology based expert system for thyroid disease diagnosis," in *Proc. Int. Conf. Nascent Technol. Eng. Field (ICNTE)*, Jan. 2015, pp. 1–6.

[20] O. Mohammed, R. Benlamri, and S. Fong, "Building a diseases symptoms ontology for medical diagnosis: An integrative approach," in *Proc. Int. Conf. Future Gener. Commun. Technol. (FGCT)*, Dec. 2013, pp. 104–108.

[21] T. Okumura and Y. Tateisi, "A lightweight approach for extracting disease-symptom relation with metamap toward automated generation of disease knowledge base," in *Proc. Int. Conf. Health Inf. Sci. (HIS)*, 2012, pp. 164–172.

[22] X. Zhou, J. Menche, A. L. Barabási, and A. Sharma, "Human symptoms–disease network," *Nature Commun.*, vol. 5, Jun. 2014, Art. no. 4212.

[23] C. Shivade, P. Malewadkar, E. Fosler-Lussier, and A. M. Lai, "Comparison of UMLS terminologies to identify risk of heart disease using clinical notes," *J. Biomed. Inform.*, vol. 58, pp. S103–S110, Sep. 2015.

[24] A. R. Aronson and F. M. Lang, "An overview of MetaMap: Historical perspective and recent advances," *J. Amer. Med. Inform. Assoc.*, vol. 17, no. 3, pp. 229–236, May 2010.

[25] S. Hwang, J. Kim, J. Kim, and S. Park, "CSNet: Constructing symptom network based on disease-symptom relationships," in *Proc. IEEE Int. Conf. Syst., Man, Cybern. (SMC)*, Oct. 2017, pp. 960–965.

[26] N. Imam and M. F. Siddiqui, "Network medicine: A step towards future medication," *J. Appl. Comput.*, vol. 2, no. 1, pp. 1–8, Feb. 2017.

[27] A. Rodríguez-González, J. Torres-Niño, M. A. Mayer, G. Alor-Hernandez, and M. D. Wilkinson, "Analysis of a multilevel diagnosis decision support system and its implications: A case study," *Comput. Math. Methods Med.*, vol. 2012, Sep. 2012, Art. no. 367345.

[28] A. Kalyanpur *et al.*, "Structured data and inference in DeepQA," *IBM J. Res. Develop.*, vol. 56, nos. 3–4, pp. 10:1–10:14, May/Jun. 2012.

[29] D. Ravi *et al.*, "Deep learning for health informatics," *IEEE J. Biomed. Health Inform.*, vol. 21, no. 1, pp. 4–21, Jan. 2017.

[30] J. A. Lossio-Ventura, C. Jonquet, M. Roche, and M. Teisseire, "Biomedical term extraction: Overview and a new methodology," *Inf. Retr. J.*, vol. 19, nos. 1–2, pp. 55–99, Apr. 2016.

[31] E. Thomas, J. Z. Pan, and Y. Ren, "TrOWL: Tractable OWL 2 reasoning infrastructure," in *Proc. 7th Int. Conf. Semantic Web, Res. Appl. (ESWC)*, 2010, pp. 431–435.

[32] S. Brin and L. Page, "The anatomy of a large-scale hypertextual Web search engine," in *Proc. 7th Int. Conf. World Wide Web (WWW)*, 1998, pp. 107–117

[33] R. Floca, "Challenges of open data in medical research," in *Opening Science*. Cham, Switzerland: Springer, 2014, pp. 297–307.

[34] L. M. Tierney, Jr., S. Saint, and M. A. Whooley, *Current Essentials of Medicine*, 4th ed. New York, NY, USA: McGraw-Hill, 2011, pp. 1–607.

[35] T. Bai, L. Gong, C. A. Kulikowski, and L. Huang, "Implicit knowledge discovery in biomedical ontologies: Computing interesting relatednesses," in *Proc. IEEE Int. Conf. Bioinf. Biomed. (BIBM)*, Nov. 2015, pp. 497–502.

[36] N. Riches *et al.*, "The effectiveness of electronic differential diagnoses (DDX) generators: A systematic review and meta-analysis," *PLoS ONE*, vol. 11, no. 3, p. e0148991, 2016.

[37] B. McFee and G. Lanckriet, "Metric learning to rank," in *Proc. 27th Int. Conf. Mach. Learn. (ICML)*, 2010, pp. 775–782.

GUN-WOO KIM received the B.S. and M.S. degrees from The University of Newcastle, Australia, in 2006 and 2007, respectively, and the Ph.D. degree in computer science and engineering from Hanyang University, South Korea, in 2017, where he is currently a Postdoctoral Researcher with the Department of Computer Science and Engineering. His research interests include e-Healthcare, personalized semantic information retrieval, and big data processing.

DONG-HO LEE received the B.S. degree from Hongik University, in 1995, and the M.S. and Ph.D. degrees in computer engineering from Seoul National University, South Korea, in 1997 and 2001, respectively. From 2001 to 2004, he was with the Software Center, Samsung Electronics Co., Ltd., where he was involved in several research projects for next-generation digital appliances. He is currently a Professor with the Department of Computer Science and Engineering, Hanyang University, South Korea. His research interests include system software for flash memory, embedded DBMS, and multimedia systems.