IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received December 10, 2020, accepted December 15, 2020, date of publication December 25, 2020,
date of current version February 18, 2021.

Digital Object Identifier 10.1109/ACCESS.2020.3047364

Exploiting Ontology Recommendation Using Text Categorization Approach

MUHAMMAD AZEEM SARWAR[1], MANSOOR AHMED[1], ASAD HABIB[2], MUHAMMAD KHALID[3,7], M. AKHTAR ALI[3], MOHSIN RAZA[4], (Member, IEEE), SHAHID HUSSAIN[5], AND GHUFRAN AHMED[6]

[1]Department of Computer Science, COMSATS University Islamabad, Islamabad 45550, Pakistan
[2]Institute of Information Technology, Kohat University of Science and Technology, Kohat 26000, Pakistan
[3]Department of Computer and Information Sciences, Northumbria University, Newcastle upon Tyne NE1 8ST, U.K.
[4]Department of Computer Science, Edge Hill University, Ormskirk L39 4QP, U.K.
[5]Department of Computer and Information Science, University of Oregon, Eugene, OR 97403, USA
[6]Department of Computer Science, National University of Computer and Emerging Sciences, Islamabad 44000, Pakistan
[7]School of Computer Science, University of Lincoln, Lincoln LN6 7TS, U.K.

Corresponding author: Mansoor Ahmed (mansoor@comsats.edu.pk)

ABSTRACT Semantic Web is considered as the backbone of web 3.0 and ontologies are an integral part of the Semantic Web. Though an increase of ontologies in different domains is reported due to various benefits which include data heterogeneity, automated information analysis, and reusability, however, finding an appropriate ontology according to user requirement remains cumbersome task due to time and efforts required, context-awareness, and computational complexity. To overcome these issues, an ontology recommendation framework is proposed. The Proposed framework employs text categorization and unsupervised learning techniques. The benefits of the proposed framework are twofold: 1) ontology organization according to the opinion of domain experts and 2) ontology recommendation with respect to user requirement. Moreover, an evaluation model is also proposed to assess the effectiveness of the proposed framework in terms of ontologies organization and recommendation. The main consequences of the proposed framework are 1) ontologies of a corpus can be organized effectively, 2) no effort and time are required to select an appropriate ontology, 3) computational complexity is only limited to the use of unsupervised learning techniques, and 4) due to no requirement of context awareness, the proposed framework can be effective for any corpus or online libraries of ontologies.

INDEX TERMS Clustering, recommendation system, semantic web, ontology, text categorization, text mining, unsupervised learning.

I. INTRODUCTION

In the digital era of living, research community is making effort to the use of information and computing technologies to manage the rapid change in data volume. The amount of data doubles over a period of 20 to 24 months [1], [2] and by 2020, the world will have over 4ZB data. The extraction of meaningful facts from the huge amount of data with minimum human involvement and effort remains a challenging task for the research community. A huge amount of data is available on the Internet and Semantic Web is considered a simple way to allow machines for precise understanding and processing of the data. Semantic Web standards enable data interoperability by constructing a distributed data space for software agents and users to access and publish information from different data sources and locations [3]. Moreover, Semantic Web standards aid applications to carry out more of the work required to find, combine, and act upon information on the web without human intervention. Ontologies are an integral part of the semantic web and bear the potential to model different types of data [4]. Ontologies provide support to share and reuse knowledge while providing automated reasoning about data. Due to structural support and formal representation of domain schemas [5], ontologies open several opportunities for researchers to automate the processing of web data such as ontology's effect on the system quality [6], heterogeneity, automated information analysis, reusability [7]. The diverse benefits of ontologies enable the

The associate editor coordinating the review of this manuscript and approving it for publication was Le Hoang Son.

research community to explore its use in certain domains such as agriculture, healthcare and information technology [8]–[10]. A frequently used ontology design principle is to reuse the content of existing similar ontologies. A huge number of ontologies are available on the Internet with the purpose of reuse instead of building it completely from scratch, such as CMS Ontology,[1] Library Ontology[2] and SNOMED CT Ontology.[3] The reuse of classes, properties, and concepts (ontology components) of an existing ontology according to a user's needs can significantly reduce the cost in terms of time and effort [10]. Due to the availability of a number of ontologies in different domains, searching for an appropriate ontology for its reuse with respect to a user requirement is considered as an ongoing challenge [11]. The existing efforts of the research community can be summarised as follows. 1) While using search engines to find the ontologies, a huge number of links and documents are provided in response to a user's query. Assistance is required to help the user find the right ontology according to the subject domain and conceptual details. Since few search engines use ranking algorithms, consequently, visibility of some well-defined ontologies can be hindered [12]. Moreover, this is in itself a tedious and time-consuming task. 2) To use recommendation systems to take keywords as input and recommend ontologies to the users, however, these recommendation systems are context-aware or domain-specific, for example, related to biomedical science. 3) Ontology libraries are used to find an appropriate ontology for users, however, few libraries support keywords search.

Like other studies [13]–[15], we consider ontology recommendation as an information retrieval problem, and employ a text categorization approach to propose a framework for the recommendation of ontologies with respect to the user requirement. Firstly, an ontology repository of four domains is created by collecting ontologies from literature and Internet resources. Secondly, a pool of requirements is created by conducting a survey from the developers and domain experts. In this regard, 31 user requirements are collected to validate the effectiveness of the proposed framework. Finally, the ontology repository and requirements pool is considered as an input to the proposed framework. The proposed framework organizes the ontologies in related groups (clusters) and whenever a user gives a requirement as input, the system will perform the required steps (mentioned in section V) and will recommend the most appropriate ontology. The proposed work overcomes the aforementioned limitations such that only a single most appropriate ontology will be provided to the user instead of a plethora of results for him to choose. Unlike ranking algorithms, the proposed system employs text categorization and unsupervised learners overcoming the issue of visibility of some well-defined ontologies. Furthermore, the proposed system eliminates the issue of domain-specific ontology search and it can be enhanced to use ontologies from many fields such as computer science, medical, education. The main contributions of this research work are as follows:

- Instead of the main repository with all ontologies, we have organized the ontologies in related groups according to the domain expert opinion. Consequently, achieving better retrieval and performance.
- An ontology recommendation framework is proposed. The proposed framework recommends the top most appropriate ontology to the user according to her requirement rather than providing with countless pages of results.
- A performance assessment model to evaluate the effectiveness of the proposed framework in organizing ontologies, predicting the correct ontology group against user requirements, and recommendation of ontology is also presented.
- We have used various weighting methods in our experimentation. We have designed experiments for determining the suitability and best weighting method out of Binary, TF, TFIDF, TFC, LTC, and Entropy for unsupervised learners.
- We have used Fuzzy c-means, K-Means (Euclidian and Manhattan), and K-Medoids for our experimentation. For each ontology group, we have identified the best-unsupervised learner in organizing and predicting the correct ontology group against user requirements.

The rest of the article is structured as follows. Section 2 presents the related work. Sections 3 and 4 describe a brief overview of ontologies and text categorization approach respectively. Section 5 describes the proposed ontology recommendation framework. Section 6 describes an evaluation model to assess the effectiveness of the proposed approach. Sections 7 and 8 present the experimental procedure and results respectively. Finally, Section 9 presents the conclusions of the proposed work.

II. RELATED WORK

This research work mainly focuses on ontology recommendation, however, we also presented the recent literature on the text categorization approach. Different researchers have made efforts to address the issue of appropriate ontology recommendation. The summary of their efforts with limitations is as follows.

A. ONTOLOGY RECOMMENDATION

Alani *et al.* [16] performed an Ontology search based on the content of ontology and user query. The authors used the query to represent domain names. Moreover, they used Web pages to find representative terms related to query for expanding the query. Experiments were performed in the biomedical domain. Jonquet *et al.* [17] introduced a web service for

[1] https://github.com/ayesha-banu79/Owl-Ontology/blob/master/College%20Mngt%20Sys.owl

[2] https://github.com/ayesha-banu79/Owl-Ontology/blob/master/Library%20Ontology.owl

[3] http://bioportal.bioontology.org/ontologies/SNOMEDCT

recommending ontologies in the biomedical domain. In this study, size, connectivity, and coverage were used for decision making for a query. The system uses a set of keywords or ontology metadata to describe the domain to recommend appropriate ontology. Martínez-Romero *et al.* [18] also proposed an ontology recommendation system for the biomedical domain known as ''BiOSS'', which recommends ontologies on the basis of keywords provided by the user. BiOSS uses domain coverage, popularity, and semantic richness as the evaluation parameters. On the basis of the scores of these parameters, ontologies are suggested in an order.

Groza *et al.* [19] proposed an ontology ranking and selection system based on the Analytical Hierarchical Process (AHP). In this study, the authors tackled the problem of selecting, evaluating, and ranking of ontologies. Moreover, the authors used AHP to analyze different ontologies from different perspectives. The system is composed of three main modules named domain coverage, ontology measurement, and AHP. The proposed system was tested on the ontologies of a the tourism domain. Butt *et al.* [20] proposed a framework entitled RecoOn for recommending ontologies based on structure-less queries. The aim of RecoOn is to suggest the best-matched ontologies against a query consisting of multiple keywords. Experiments were conducted on the CBRBench ontology collection. Matching cost and popularity of the ontology was used as the evaluation metrics.

Trokanas and Cecelja [21] proposed an algorithm for ontology evaluation and reuse. The proposed algorithm uses knowledge about ontologies, which is presented in the form of terminologies and structure to create the compatibility metrics. The algorithm relies on the high-level details of ontology. Chemical and business process use cases have been used for demonstrating the work. Aguilar *et al.* [22] proposed a hybrid recommender system for ontologies of the biomedical domain. The authors used metadata, which is stored for ontologies in the semantic repository, and considered quality and adaptability characteristics of ontologies during the recommending process.

Brown *et al.* [23] used the concept of ontology recommendation in recommending the ontologies for the planning of software requirements. The process is divided into two phases. In the first phase requirement model is converted into ontology. In the second phase, converted ontology is compared to the other ontologies related to the domain. A tool was developed for the second phase that consists of three components such as Matchmaker, persistence manager, and query handler. Recommended ontologies are determined on the basis of these three components.

Zulkarnain *et al.* [24] proposed a methodology by using reuse, coverage, language as acceptance criteria for ontology recommendation. To verify the ontology recommender system, the authors used the BioPortal's ontology recommender's API. The recommended bio ontology can be further reused and enhanced according to the need. Martínez-Romero *et al.* [25] extended their previous work [18] and proposed an ontology recommendation system called ''NCBO Ontology Recommender 2.0'' for recommending biomedical ontologies. The proposed system finds ontologies based on biomedical text or keywords using coverage, detail, acceptance, and specialization as evaluation parameters. ''NCBO Ontology Recommender 2.0'' recommends more than 500 ontologies available on NCBO BioPortal. Faessler *et al.* [26] proposed JOYCE, a tool for selecting and tailoring ontologies. JOYCE identifies and assembles ontologies or pieces of ontologies from the ontology repository. The aim of the proposed tool is to utilize the existing ontologies.

Finally, the related work of ontology recommendation and its limitations are summarized in Table 1. Besides, there are some ontology libraries and search engines available. However, mostly they are domain-specific and their scope is limited. The main limitations of existing efforts for ontology recommendation are: 1) Context-awareness, 2) limited scope, 3) efforts required to implement the conceptual models, and 4) use of single or multi-term keywords. Considering user requirement description as an input of an ontology recommendation system can improve the searching process regardless of the existence of numerous ontologies of any domain. As we are considering ontology recommendation as an information retrieval problem, a brief overview of related literature is presented in the next section.

B. TEXT CATEGORIZATION

Traditional machine learning-based approaches for text categorization primarily focuses on feature engineering and classification of text documents. Machine learning models take text features as input, which are designed based on several statistical methods and word frequency. Several domains have benefited from machine learning and text categorization approaches. Hussain *et al.* in [27], have employed machine learning and text categorization approach for automating the selection of design pattern. The proposed three-step methodology contains pre-processing, unsupervised learning of identifying similar objects and selection of appropriate design patterns. The authors evaluated the performance of their proposed system and reported 18% better performance as compared to supervised learners. The authors extended their research to include a large dataset of design patterns, employed several statistical methods of text features and unsupervised learners [28]. Compared to previous work, the extended approach provided four advantages. Firstly, the semi-formal definition of design patterns was not necessary as a prerequisite; secondly, the ground class labels assignment was not mandatory; thirdly, the lack of classification training for each design pattern class, and fourthly, authors claimed that appropriate sample size was not needed for accurate training. Authors in [29], proposed a framework for selection and organization design patterns. The authors tried on to minimize the semantic relationship gap between design patterns and the features. The authors presented a case study and employed a powerful deep learning algorithm named Deep Belief Network.

TABLE 1. **Related work.**

Ref.	Problem Addressed	Implementation tools	Evaluation Parameters	Ontology dataset	Input	Limitations
Alani *et al.* [16]	Searching of ontologies.	3Store SPARQL.	Precision Recall F-measure.	55 Open Biomedical Ontologies.	Multi-term keyword.	Domain Specific
Jonquet *et al.* [17]	Ontology recommendation.	REST web service.	Coverage Connectivity Size.	Ontologies from NCBO portal and UMLS Meta-thesaurus.	Set of keywords Textual metadata.	Domain Specific
Martínez-Romero *et al.* [18]	Recommendation of ontologies.	Java Apache Axis 2 SOAP protocol Microsoft .NET.	Domain coverage Popularity Semantic richness.	200 biomedical ontologies.	Keywords.	Domain Specific
Groza *et al.* [19]	Ontology selection and ranking	Visual priority estimation. Tool PriEsT for AHP implementation.	Size Consistency Cohesion Domain coverage.	17 tourism domain ontologies.	user-given terms	Domain Specific Size of ontology dataset.
Butt *et al.* [20]	Recommendation of ontologies based on structure-less queries.	NA	Matching cost Popularity.	CBRBench ontology collection.	Multi-term query keyword string	Keyword based
Trokanas *et al.* [21]	Ontology evaluation and reuse for the domain of Process Systems Engineering.	NA	Natural language level, Encoding information, External resources, Size, and breadth. Cosine similarity and Euclidean distance.	Conference ontologies and three eSymbiosis ontologies.	NA	Experimentation involved only three esymbiosis ontologies.
Aguilar *et al.* [22]	Recommendation based on content and Collaborative recommendation of ontologies.	NA	NA	4 ontologies related to biomedical.	NA	Size of dataset.
Brown *et al.* [23]	Ontology selection in software requirement planning.	Protégé, JENA, Sesame.	Similarity level using semantic matchmaking.	Set of ten random, Non-related domain ontologies.	Requirement model.	More focused on requirement planning.
Zulkarnain *et al.* [24]	Methodology for reusing Biomedical Ontologies.	NCBO Portal's ontology recommendation API.	Detail, Acceptance, Specialization.	BioPortal's ontologies.	Text keywords.	Domain Specific.
Martínez-Romero *et al.* [25]	Recommendation of ontologies.	Ruby on Rails Web Framework, JavaScript language Ruby language.	Coverage, Detail, Acceptance Specialization.	More than 500 ontologies on NCBO Bio Portal.	Biomedical Text Biomedical keywords.	Domain Specific
Faessler *et al.* [26]	A scalable tool for identification and assembling of relevant ontologies.	Jena	Coverage, overlap, and overhead.	323 BioPortal's ontologies.	Keywords.	Domain Specific.

Several other studies have also employed machine learning bases solutions for text categorization in different domains. In [30], a spam detector is developed using machine learning. The proposed solution uses a combination of a collection of features, pre-processing steps, or setup information, such as using or not using stop words list, lemmatization, keyword patterns, etcetera (etc). Vilares *et al.* [31] present an unsupervised approach to multiple languages sentiment analysis guided by rules based on syntax; the terms are weighted based on the syntax-graph analysis. Text categorization approaches are also been tested on many languages related text corpuses such as the Turkish Language [32], Arabic language [33], and Croatian Language [34]. Author profiling is yet another significant task relevant to the categorization of text, where a lot of progress has been observed. In this regard, Basile *et al.* [35] have proposed an author profiling model. The proposed models consist of a linear kernel SVM, Parts Of Speech (POS), and n-grams.

Another area of study is collaborative filtering, hash collaborative filtering, and binary codes for the recommendation systems. Collaborative filtering algorithms recommend the items to a user, based on the preferences of the customer and are able to match other users with common interests [36]. Binary codes aim to approximate user-item encounters and create hash tables to speed up retrieval time. Using binary codes can reduce the query time to constant or sublinear complexity considerably. By learning binary codes, the storage requirement can be minimized considerably, as storing each binary code needs just 4 bytes if the code length is 32 [37]. Various studies have been conducted to evaluate the usage of binary codes in e-commerce, to recommend individual items to users [36], [39], [39]–[42][38]–[42] and personalized fashion recommendations [43]. The accuracy of these models is lower than traditional models because such models are highly limited and can lack adequate versatility to list the Top-N objects correctly [39].

In recent years, state-of-the-art approaches have moved dramatically from computational such as statistical and traditional machine learning to deep learning-based text categorization [44]. Convolutional Neural Networks (CNN) are commonly employed in the field of image processing. Vieira and Moura [45] introduced the application of CNN in text categorization. For several classification datasets, Kim employed a single layer CNN achieving impressive

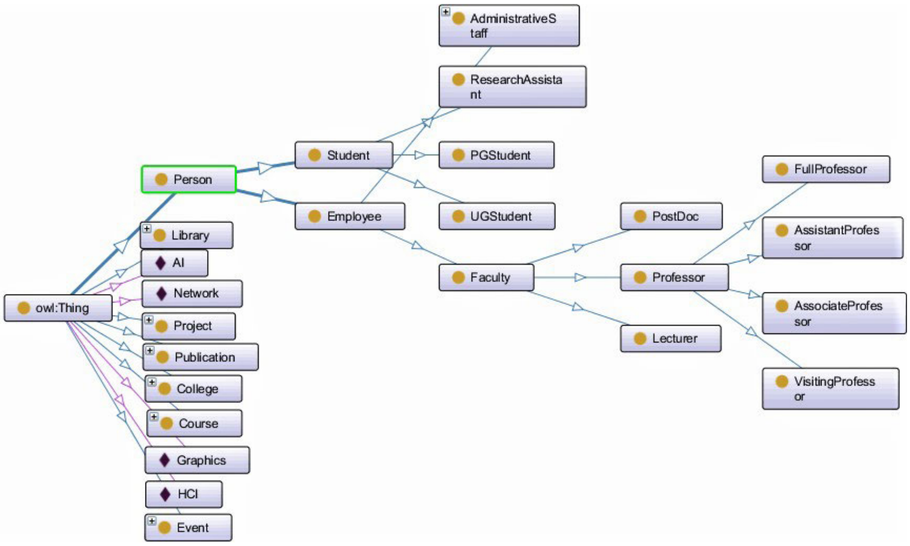


FIGURE 1. Graphical representation of an institute ontology.

classification results. In [46], Liu *et al.* employed the Recurrent Neural Network (RNN) for text categorization. Unlike previous works, the authors focused on multitask learning system to learn together across several related tasks. To enhance the efficacy of deep learning models to accurately classify the text several authors used the combination of two models. The authors in [47] proposed a text classification model comprising Long Short Term Memory Network (LSTM) and CNN. LSTM is also being used in the field of healthcare. Authors in [48] and [49] used LSTM and deep learning respectively for intelligent healthcare monitoring systems. In another attempt, authors employed a combination of CNN and RNN [50]. CNN is used to extract text features while RNN is responsible for multi-label prediction.

III. BRIEF OVERVIEW OF ONTOLOGIES

In order to imply different viewpoints, several people, software programs, and organizations communicate with each other despite their differences in needs, platforms, formats, and backgrounds [51]. An ontology consists of a set of terms that are used in a formal and hierarchical manner to constitute ontology. These terms include class, subclass, properties, and individuals. In this regard, the term ontology can be described as “a hierarchically structured set of terms to describe a domain that can be used as a skeletal foundation for a knowledge base” [52].

Being the backbone of the Semantic Web, ontologies are regarded as an alternative to address data heterogeneity problems. The term ontology is defined as “a formal, explicit specification of a shared conceptualization” [20].

Ontologies consist of concepts or objects that can be used to express knowledge and relationships [52]. A concept can be any real-world object. There are no strict rules to describe the term concept in the ontology. However, a concept should reflect the same real-world phenomena that a specific ontology is expressing. An ontology consists of a set of elements that are used in a formal and hierarchical manner to constitute ontology. The ontology has four primary elements: classes, concepts, instances, and relationships [53]. Creating an ontology also promotes the analysis of knowledge in the domain, which in effect helps to reuse existing ontologies [55]. The graphical representation of ontology to model the concept of an institute is shown in Fig 1. This ontology contains 41 classes and 42 sub-classes. For example, the person class contains two subclasses named student and employee. Moreover, Employee class contains two subclasses named administrative staff and faculty creating a hierarchy of different concepts as they appear in the real world

Ontologies can be used in many research areas to support a wide range of tasks such as natural language processing, knowledge representation, information retrieval, databases, online database integration, knowledge management, visual information retrieval, geographic information systems, digital libraries, or multi-agent systems [56]. Furthermore, many researchers are using the ontology related systems in different fields such as Diagnostics [57], Recommendation and classification [58], [59], IoT security [60], content analysis [61] and opinion mining [62]. However, considering the ontology reuse as a defined design pattern, little or no attention is being paid to the reuse of existing ontologies to reduce the costs [10]. Consequently, reuse and discovery of ontology terms remain a crucial challenge. To address these issues we are proposing a framework to recommend appropriate ontology to the users on the basis of user requirement.

IV. BRIEF OVERVIEW OF TEXT CATEGORIZATION APPROACH

The rapid increase in the amount of digital information available on the internet has made it difficult to search for relevant information for a user. Consequently, the categorization of documents has become a challenging task; it enables researchers to consider it as an information retrieval problem. The research community has reported the implication of the text categorization approach to address information retrieval problems [63]. Text categorization is a process that analyses given electronic documents algorithmically and assigns them to related categories [64].

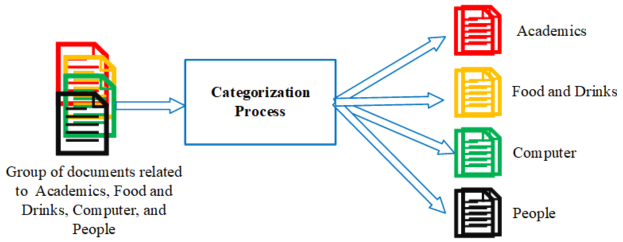

Group of documents related to Academics, Food and Drinks, Computer, and People
Categorization Process
Academics
Food and Drinks
Computer
People

FIGURE 2. Basic work of text categorization.

Fig 2 shows the graphical representation of the basic work of the automated text categorization approach. Automatic text categorization is used in machine learning, especially in the text-mining domain, which employs either unsupervised or supervised learning techniques. Supervised learning techniques require that class labels must be assigned to documents, whereas unsupervised learning techniques use data attributes, similarity and dissimilarity measure to automate their learning process. A conventional text categorization framework involves pre-processing of text documents, feature extraction, feature selection, and classification of these documents [65].

A. PRE-PROCESSING

For the text categorization of large documents, it is necessary to perform pre-processing of the input documents and store the extracted information in an appropriate data structure for further steps [66]. The pre-processing step involves tokenization, removal of stop-words, lowercase conversion, and stemming (or lemmatization). *Tokenization* is a process splitting a text stream into single words, phrases or any other meaningful parts. *Stop-words* removal process discards frequent words that carry no meaning or information, such as propositions, pronouns, and conjunctions. Subsequently, tokens are converted into lowercase to reduce duplication of words. *Stemming* is a process for performing *word normalization*, which reduces a word into its basic form [67].

B. INDEXING

Indexing is the common way to convert textual documents into numeric vectors. Vector Space Model (VSM) is employed as the most common indexing method to describe a document in a numeric vector. Regardless of its simple data structure, the VSM enables efficient analysis of large document collections. VSM was originally introduced for indexing documents and retrieval of information. However, it is now being used in different text mining and document retrieval systems [68]. We have used the *term document category* of VSM in this study, where each word is represented by a numeric value demonstrating the importance (weight) of the word in a document. Equation 1 is used to construct VSM (word-by-document matrix) where entry of each word refers to its occurrence in the document.

$$D = (W_{wd}) \tag{1}$$

In Equation 1, D denotes the word-by-document matrix, whereas W corresponds to the weight of the word w in document d. Documents are presented as vectors of terms in the VSM to be processed by classifiers. Different terms in VSM have distinct degrees of meaning that signify a document's semantics. Term weighting schemes are commonly used in document representation to improve text-categorization. In this regard, various weighting schemes that give appropriate weights to terms are being used by the research community. We considered most commonly used methods such as *Entropy Weighting*, *Term Frequency-Inverse Document Frequency* (TFIDF), *Length Term Collection* (LTC), *Term Frequency Collection* (TFC) and *Binary* [66]. A brief overview of each weighting method used is as follows:

1) BINARY

Binary weighting method is considered as the simplest weighting method. As the name suggests, if a word occurs in the document the weight will be 1 and if the word does not occur then the weight will be 0.

2) TFIDF

TFIDF is a numerical metric designed to represent how significant a word is to a document in a list or corpus. TFIDF value decreases proportionally to the amount of times a word occurs in the document and is determined by the amount of documents in the corpus containing the term, which helps to account for such terms appearing more often overall.

3) ENTROPY

Entropy is a measure of the unpredictability or imbalance. The entropy word weight characterizes a word's value in identifying a specific document. When a word occurs specifically in a document then entropy is high and if the word appears equally in the documents then the weight (entropy) is low.

4) TFC

The TFIDF weighting method does not take into account the length of documents. TFC is a variant of TFIDF, however, for TFC length normalisation is used. TFC uses a normalized TFIDF weight for document terms.

5) LTC

LTC is also a different format of TF-IDF like TFC. However, it considers the limit of small datasets, and normalization of weights. Furthermore, instead of the raw word frequency,

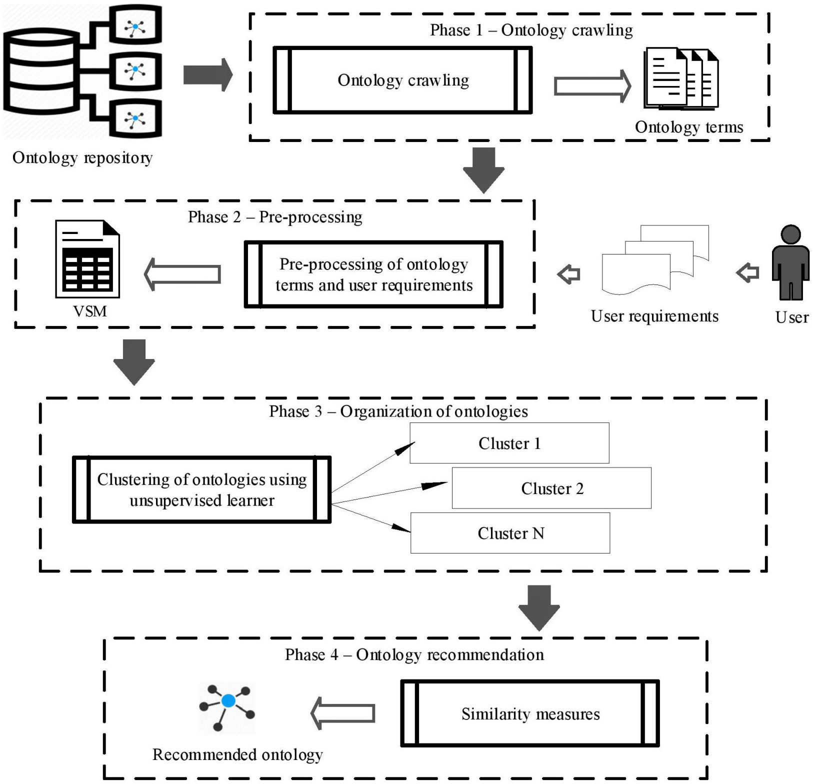

Ontology repository
Phase 1 – Ontology crawling
Ontology crawling
Ontology terms
Phase 2 – Pre-processing
VSM
Pre-processing of ontology terms and user requirements
User requirements
User
Phase 3 – Organization of ontologies
Clustering of ontologies using unsupervised learner
Cluster 1
Cluster 2
Cluster N
Phase 4 – Ontology recommendation
Recommended ontology
Similarity measures

FIGURE 3. Proposed system methodology.

LTC uses the logarithm of the word frequency, thereby minimising the impact of large frequency variations.

The main idea behind using these weighting methods in this research is to increase the accuracy of text categorization and find the best-fit weighting method for ontology corpus. Consequently, while proposing an ontology recommendation system we also performed a comparative study of feature weighting methods. Five aforementioned methods were evaluated on ontology corpus of four domains with three unsupervised learners. Three random terms are selected from our corpus and their consequent feature values are presented in Table 2 for better understanding of readers. It can be seen in the Table 2 that how different weighting schemes treat various terms in the VSM (depending on the length of document and frequency of the term) and assign weights to them. For a detailed explanation about various weighting methods and their implications, readers can refer to [69]–[71].

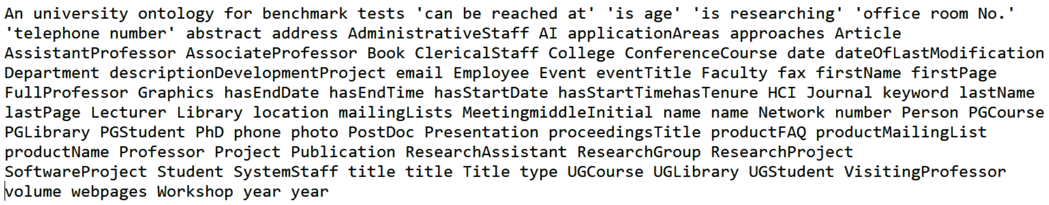
V. PROPOSED METHODOLOGY

The selection of appropriate ontology with respect to the user requirement has become a complex process in terms of required time and effort. The research community has reported different implications of machine learning and text categorization in several domains such as author identification [72], web page classification [73], spam e-mail filtering [74], sentiment analysis [75] and design pattern classification and recommendation [28]. Although several frameworks and statistical methods have been introduced for ontology detection with respect to the given user keywords and queries, however, to the best of our knowledge, there is presently no comprehensive study on implications of text categorization approach in terms of ontology recommendation.

In this study, we propose a framework that employs unsupervised learning and text categorization approach for ontology recommendation. The objectives of the proposed framework are: 1) to organize ontologies according to the opinion of domain experts, and 2) to select appropriate ontology with respect to the user requirement. The layout of the proposed recommendation framework is shown in Fig 3, which describes its functionality in four phases. In the first phase, ontology crawling is designed and implemented to obtain ontology terms and text. Subsequently, in the second phase, pre-processing activities are performed over user requirements (in natural language) and ontology data. In the third phase, unsupervised learning is employed to group similar ontologies and determine candidate ontology group for the user requirement being processed. Finally, in the fourth phase, ontology is suggested for the given user requirement.

TABLE 2. Random Terms and Corresponding Weighting Methods Values.

Term	Binary	Tf-idf	Entropy	LTC	TFC
address	1	0.6493	0.6758	0.2569	0.1636
agent	1	1.8325	1.0711	0.5746	0.4619
application	1	1.3581	0.6782	0.5373	0.3423



```
An university ontology for benchmark tests 'can be reached at' 'is age' 'is researching' 'office room No.'
'telephone number' abstract address AdministrativeStaff AI applicationAreas approaches Article
AssistantProfessor AssociateProfessor Book ClericalStaff College ConferenceCourse date dateOfLastModification
Department descriptionDevelopmentProject email Employee Event eventTitle Faculty fax firstName firstPage
FullProfessor Graphics hasEndDate hasEndTime hasStartDate hasStartTimehasTenure HCI Journal keyword lastName
lastPage Lecturer Library location mailingLists MeetingmiddleInitial name name Network number Person PGCourse
PGLibrary PGStudent PhD phone photo PostDoc Presentation proceedingsTitle productFAQ productMailingList
productName Professor Project Publication ResearchAssistant ResearchGroup ResearchProject
SoftwareProject Student SystemStaff title title Title type UGCourse UGLibrary UGStudent VisitingProfessor
volume webpages Workshop year year
```

FIGURE 4. Text corpus of an institute ontology.

TABLE 3. Details of various ontology categories and sub-categories.

Domain	Sub-Domain/Category	Number of ontologies	Total ontologies	Number of non-repeated words
Academics	Research bibliography	8	21	692
	Educational institute	5		
	Books	8		
People	Work-related	6	18	351
	People's contacts	3		
	Family history	9		
Computer Science	Networking	13	35	1540
	Cybersecurity	8		
	Software systems	11		
	Sentiments/emotions	3		
Food and Drinks	Food	12	21	1206
	Drinks	9		

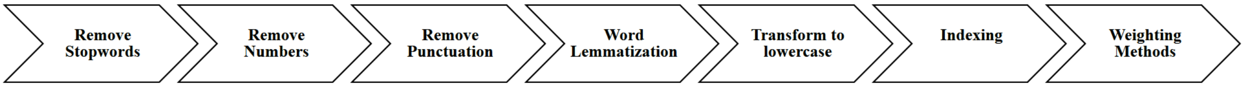


FIGURE 5. Activities involved in pre-processing.

A. ONTOLOGY CRAWLING

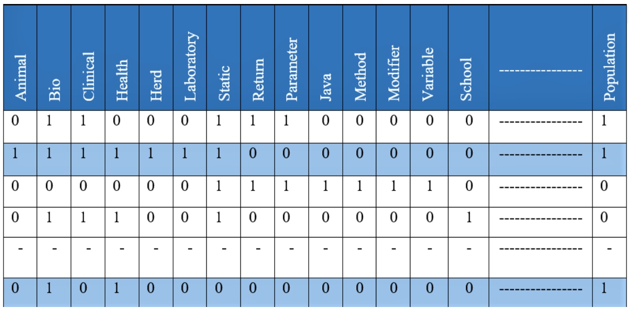
An ontology describes the information in semi-structured natural language text [76]. In other words, an ontology models any described concepts (also called classes) in terms of task, action, function, reasoning process, and strategy [77]. The aim of the crawling phase is to retrieve properties, classes, annotation properties, and metadata descriptions of ontologies and to create a text corpus for further processing. These ontologies are related to four different domains: Food and Drinks, Academics, Computer Science, and People. Consequently, for each ontology repository, a new separate file is created to represent the description of ontology. Fig 4 presents text corpus of an institute ontology after the ontology crawling step. Text corpus file contains all the classes, properties, metadata descriptions of aforementioned ontology. Table 3 presents an overview of Domains, sub-domains/categories used in this study. We used ''owlready'', a python library to extract properties, classes, annotation properties, and metadata descriptions of ontologies. Consequently, these properties, classes and descriptions are then given as input to pre-processing activity.

B. PRE-PROCESSING

The aim of the second phase is to pre-process the text data retrieved from the ontologies corpus. Pre-processing prepares the data for the next phase (clustering). The set of pre-processing activities are shown in Fig 5. The first three pre-processing activities are performed to remove stopwords, numbers, and punctuation, which have no meaning. The *word lemmatization* activity is performed to group several inflected forms of a word into a single item. Subsequently, to avoid duplication of the words due to the upper or lower case, all words are converted into lowercase. Moreover, the aim of these activities is to reduce the data sparsity and feature set size. The next activity *word indexing* constructs VSM, which contains words of all the input documents, and represents them as the word-by-document matrix. Subsequently, a feature vector is generated for each ontology. Finally, the aim of the last pre-processing activity namely *weighting methods* is applied to rank the words in VSM. For each ontology group, we determine the best performer out of the five weighting methods (Entropy Weighting, TFIDF, LTC, TFC, and Binary). The binary-weighted form of a VSM is shown in Fig 6. Furthermore, Table 3 presents the number of non-repeated words which were obtained after performing pre-processing activity on each ontology group.

C. CLUSTERING

Clustering, also known as learning without a teacher (unsupervised learning), has been applied in a wide range of fields including engineering, informatics, computer science, life and medical sciences, economics, earth sciences, and social sciences [78]. There are several clustering algorithms such as K-Medoids, K-Means, Agglomerative, Fuzzy c-means and so on. Based on the clustering properties, these algorithms



Animal	Bio	Clinical	Health	Herd	Laboratory	Static	Return	Parameter	Java	Method	Modifier	Variable	School	----------------	Population
0	1	1	0	0	0	1	1	1	0	0	0	0	0	----------------	1
1	1	1	1	1	1	1	0	0	0	0	0	0	0	----------------	1
0	0	0	0	0	0	1	1	1	1	1	1	1	0	----------------	0
0	1	1	1	0	0	1	0	0	0	0	0	0	1	----------------	0
-	-	-	-	-	-	-	-	-	-	-	-	-	-	----------------	-
0	1	0	1	0	0	0	0	0	0	0	0	0	0	----------------	1

FIGURE 6. Vector space model for indexing of ontologies with binary weighting method.

Algorithm 1 Working of K-Means

```
  Input: k: number of clusters, D: dataset
  Result: k Number of clusters
1 initialization;
2 if $k == 1$ then
3     Exit;
4 else
5     Take K distinct points randomly. These points act as
      initial centroids.
6     Assign each data object to the group most close to
      the centroid.
7     When all data objects are assigned, recalculate the
      positions of the k centroids.
8     Repeat Step 6 and 7 until the convergence is reached
      (centroids no longer move and are fixed).
9 end
```

can be grouped into certain schemes such as partitioning, hierarchical, model-based, grid-based, density-based and soft-computing [28]. The research community has agreed that no single unsupervised learning algorithm can be recommended as an outperformed learner. Like [14], we used K-Means, K-Medoids, and Fuzzy c-means to employ the proposed framework for ontology recommendation with respect to user requirements. The brief description of these algorithms (*unsupervised learners*) is as follows.

1) K-MEANS

K-Means is a renowned partitioning based iterative clustering algorithm. K-Means classifies the given data into different groups (clusters) using the idea of centroid [79]. In a cluster, the mean value of its data points is known as *centroid*. For each data vector, K-Means calculates the distance between the data vector and each cluster centroid. For any given data set, the algorithm classifies the dataset into a user-defined number of clusters namely *k*. The working procedure of the K-means algorithm is presented in Algorithm 1.

2) K-MEDOIDS

Like K-Means, K-Medoids is also a clustering algorithm based on partitioning. But, K-Medoids is more robust than K-Means [80]. In the K-Medoids algorithm, *medoids* are the data objects of clusters which are located centrally and selected randomly from the data objects *D* to form *k* clusters. Moreover, the rest of the data objects in *D* are placed near to Medoids (central point) in a cluster. Subsequently, all data objects of a cluster are processed to find new Medoids in repeated fashion and represent a new cluster in a better way. After each iteration, the location of Medoids is changed. We used K-Medoids with Euclidean and Manhattan distance. The working procedure of the K-Medoids algorithm is presented in Algorithm 2.

Algorithm 2 Working of K-Medoids

```
   Input: k: number of clusters, D: dataset
   Result: k Number of clusters
1  initialization;
2  if $k == 1$ then
3      Exit;
4  else
5      Randomly select k as the Medoid for n data points.
6      By calculating the distance between Medoid k and
       data points, Find the closest Medoid and map data
       objects to that Medoid.
7      foreach (data point o Associated to Medoid m) do
8          Swap m and o to compute the total cost of the
           configuration then select the Medoid o with the
           lowest cost of the configuration.
9      end
10     If there is no change in the assignments repeat steps
       5 and 6 alternatively.
11 end
```

3) FUZZY C-MEANS

Fuzzy c-means is a clustering algorithm that allows one data point to belong to different clusters, whereas K-Means only assign one data point to one cluster. Fuzzy c-means was introduced by Dunn in 1973 and later Bezdek improved it in 1981 [81]. The main objective of using these unsupervised learners is to classify the data objects into clusters. Fuzzy c-means works by assigning a membership value to each data point, which corresponds to each cluster center on the basis of the distance between the center of the cluster and the data point. Fuzzy c-means classify objects into clusters based on the membership function, which represents its fuzzy behavior. The membership function of Fuzzy c-means produces membership degree values, which range between 0 and 1. The working procedure of the Fuzzy c-means algorithm is presented in Algorithm 3.

D. ONTOLOGY RECOMMENDATION

The last phase of our proposed approach recommends ontology to the user on the basis of their requirement description.

Algorithm 3 Working of Fuzzy c-Means

```
Input: k: number of clusters, D: dataset
Result: k Number of clusters
initialization;
if $k == 1$ then
    Exit;
else
    Input the dataset and value of k.
    Calculate the fuzzy membership matrix
    Compute the fuzzy centers.
    Update the membership value.
end
```

When input documents are represented in the form of term vectors, the similarity between two documents is computed through their correlation. In order to suggest the appropriate document for a given requirement, similarity measures such as Dice Coefficient, Pearson Correlation, Cosine, and Extended Jacquard can be used [28]. However, for our proposed approach, we used a well-known similarity method named *Cosine Similarity* (CS), which helps to measure correlation between different vectors regardless of the document length. Moreover, it performs better than any other similarity method in text clustering [82]. The CS between two vectors calculates the cosine of the angle between these vectors.

VI. EVALUATION MODEL FOR THE PROPOSED APPROACH

In this section, we propose an evaluation model to assess the performance of the proposed framework. Firstly, we suggest measures to 1) to determine the best weighting method, 2) to evaluate the performance of unsupervised learning techniques namely K-Means, Fuzzy c-means and K-Medoids (Euclidian and Manhattan) in terms of organizing ontologies, and 3) to determine candidate ontology for the particular user requirement description using unsupervised learners. Secondly, we suggest measures to evaluate the performance of the proposed framework in the recommendation of ontology for a user requirement. Thirdly, we describe the ontology corpus and related user requirement, which are considered in four case studies.

A. EVALUATION OF UNSUPERVISED LEARNERS AND BEST WEIGHTING METHOD

The formation of clusters is an important process. However, it is also important and meaningful to test the accuracy and validity of the formed clusters. There are several measures that are used to evaluate the performance of the three clustering algorithms. We used, however, the widely used evaluation measures, which are Rand Index (RI), V-measure, Accuracy, F-measure, Adjusted Rand Index (ARI), Precision, and Recall [28], [81]. The effectiveness of an unsupervised learning algorithm depends on the higher value of these metrics. In this study, we used One-vs-All (OVA) matrix method to compute the average accuracy for two purposes: 1) to select the best weighting method for each unsupervised learner on a target ontology corpus, and 2) to identify the outperformed unsupervised learner. The reason behind using the OVA measure is due to its wide application for the multi-class problem. OVA considers the performance of the algorithm with respect to one class at a time before averaging the metrics [83]. Moreover, we use OVA for evaluating: 1) the best weighting method, 2) performance of unsupervised learners in terms of organizing ontologies, and 3) candidate ontology category for a particular requirement description.

B. EVALUATION OF ONTOLOGY RECOMMENDATION

Firstly, we suggest and apply the CS measure which is applied in order to recommend an appropriate ontology to the user. For evaluating the overall performance of the framework in recommending the ontologies, we used the Ratio of Correctly predicted Ontology *(RCO)* for the user requirements. The value of *RCO* can be computed using Equation 2 where *CSO* is correctly suggested ontologies and *SO* is suggested ontologies.

$$\text{RCO} = (\text{Number of CSO})/(\text{Total Number of SO}) \tag{2}$$

C. ONTOLOGY CATEGORIES

Numerous researchers and Semantic Web experts of different domains (such as security and privacy, e-commerce, health, bio, and so on) have developed several ontologies for the sake of their work and for motivating the reuse of ontologies, hence, a plethora of ontologies are available online. In this study, we formulated an ontology repository consisting of 95 ontologies of four domains: computer science, food and drinks, people and academics. For each domain, the collected ontologies are grouped into certain sub-categories. Details of domains, sub-domains/categories and number of ontologies are presented in Table 3. We gathered these ontologies from literature and the Internet. The brief introduction and descriptive statistics of each ontology domain are as follows.

1) COMPUTER SCIENCE

The computer science domain contains 35 ontologies, which are further divided into four categories: *networking*, *cybersecurity*, *software systems* and *sentiments/emotions*. Sentiment/Emotion group contains application specific ontologies related to emotions and sentiments analysis such as OntoSenticNet [84]. There are 13, 8, 11 and 3 ontologies in the categories of networking, cybersecurity, software systems, and sentiments/emotions, respectively. This case study includes 1540 non-repeated words of 35 ontologies after performing the pre-processing activities (Section V-B).

2) FOOD AND DRINKS

The Food and Drinks domain contains 21 ontologies, which are further divided into two categories: *Food* and *Drinks*. The Food category contains 12 ontologies related to eatable items such as pizza, ingredients, and recipes to make food. Subsequently, the Drinks category contains 9 ontologies related to different drinks such as wine, beer, coffee and so on. This case study includes 1206 non-repeated words of 21 ontologies after performing the pre-processing activities (Section V-B).

3) PEOPLE

The People domain contains 18 ontologies, which are further divided into three categories: *Work-related*, *People's contacts*, and *Family hierarchy & history*. The Work-related category consists of 6 ontologies; the People's contacts category contains 3 ontologies; the Family hierarchy & history category contains 9 ontologies. This case study includes 351 non-repeated words of 18 ontologies after performing the pre-processing activities (Section V-B).

4) ACADEMICS

The Academics domain contains 21 ontologies, which are further divided into three categories: *Research & bibliography*, *Educational institute*, and *Books*. Research & bibliography category contains 8 ontologies; the Educational institute category contains 5 ontologies; the Books category contains 8 ontologies. This case study includes 692 non-repeated words of 21 ontologies after performing the pre-processing activities (Section V-B).

D. USER REQUIREMENTS

In order to test the validity of the proposed framework in recommending ontologies, we test the accuracy of the system in recommending ontology based on user requirements. We involve a cohort of graduate students who studied the Semantic Web course. We trained the cohort and collected 31 requirements related to the above four ontology domains. We also involve three domain experts of Semantic Web and ontologies in order to identify the ontologies for the 30 requirements. Each expert identified 30 (out of 92) most appropriate ontologies for the given user requirements. In this section, we only provide a description of 13 user requirements (four for computer science group and three for the rest of ontology domains defined in Section VI-C) to evaluate the effectiveness of the proposed framework.

1) REQUIREMENTS OF THE COMPUTER SCIENCE DOMAIN

The user requirements (UR) of the computer science domain are given as follows.

UR- 1: "I need ontology so i can report the bugs of software. This ontology must have bug type, bug report, bug status, and report status report contains priority, severity, and report attributes. It should also provide a solution and fixed version and it generates a summary of the bug report, the ontology should also provide information such as bug is resolved by person or community."

UR-2: "An ontology is required for cyber systems. Mainly the ontology should focus on attack pattern detection. The ontology should contain the taxonomy of problems and concepts related to the cyber world, E.g. weakness and vulnerability of system, target, probing techniques, impact of attack, and types of attacks, attack steps, patterns, technique and description This ontology provides a vocabulary and representation for the Common Attack Pattern Enumeration and Classification (CAPEC) which provides a publicly available, community-developed list of common attack patterns along with a comprehensive schema and classification taxonomy. Attack patterns are descriptions of common methods for exploiting software systems. They derive from the concept of design patterns applied in a destructive rather than constructive context and are generated from in-depth analysis of specific real-world exploit examples."

UR-3: "An ontology that contains all the pieces of the configuration of a server, the ontology should have concepts ranging from the server implementation to the user database and the policy being maintained by the server. Moreover, basic authorization and authentication manager should be there for security reasons."

UR-4: "I am building an ontology based system for sentiment analysis so I need an ontology related to the sentiment analysis. Ontology must contain all behaviors or emotions such as happy, sad, angry, uncomfortable, pain etc."

2) REQUIREMENTS OF THE ACADEMICS DOMAIN

The URs of the academics domain are given as follows.

UR-5: "Ontology for university benchmark is required. It contains information of faculty such as dean, director, Full Professor, Clerical Staff, Professor, lecturer, teaching staff and students. This ontology contains information of Graduate Course, research articles and publications. Against each publication and research, article data is stored such as research interest, title, Publication author ad publication year."

UR-6: "Academic ontology that must contain the information conference, journal, and author. Organization name and author name are provided. It also must have the year of conference and journal. Conference name and journal name are provided. Publications of the author must be provided in the ontology. This publication must contain some pub id.it contain pages and titles of conference and journal."

UR-7: "Ontology that contains university information. It contains faculty such as associate professor, full professor, researcher, teachers, and external teacher. It includes courses and these courses are taught by some teacher. Last name and first name of each faculty and student are provided. Titles of courses are part of this university information system."

3) REQUIREMENTS OF FOOD AND DRINKS DOMAIN

The URs of the food and drinks domain are given as follows.

UR-8: "This ontology contains the detail of all coffees. Base, drink, and a topping of coffees must be defined properly. It contains the ingredients of the coffee like condensed milk, stream milk, level of sugar and water."

UR-9: "Food ontology that models the ingredients of pizza and provides the vegetable ingredients that are used to make pizza. It contains the types of pizza ad its sizes such as large pizza, small pizza, meat only pizza, medium pizza, and vegetarian pizza. This ontology also contains ingredients for meat only pizza.''

UR-10: ''Ontology that model the cocktails, drinks, and beverages. This ontology describes the ingredients of the drinks and cocktails. Hot sauces and Worcestershire sauce are used to serve the beverages and drinks. Alcoholic and nonalcoholic beverages must be included in this ontology separately such as brandies and rums, coconut milk and coffees.''

4) REQUIREMENTS OF THE PEOPLE DOMAIN

The URs of the people domain are given as follows.

UR-11: ''We are developing a system that involves keeping a tight record of bio-data of people. In this regard, we need an ontology that we can align with an ontology that we are developing. The ontology should contain complete contact details of a person involving Address, country, state, cell phone number, email address etc.''

UR-12: ''We are developing a system that involves keeping a record of a person's family. In this regard, we need an ontology that we can align with the ontology that we are developing. The ontology should contain complete family details of a person's children, spouse, parents, etc. moreover, according to a person's gender the close relations he has i.e. aunt, nephew, niece etc.''

UR-13: ''An ontology is required containing concepts related to artists and their prominent works and their early life details (e.g. born, school, died etc.).''

VII. EXPERIMENTAL PROCEDURE

In this section, we describe the tools used to perform the experiment. Moreover, for evaluating the proposed approach, we devised three pseudocodes for experimental procedures, which are: 1) organize ontologies, 2) determine appropriate ontology domain, and 3) select the most appropriate ontology.

A. TOOLS USED IN EXPERIMENTATION

We performed all the experimentation process on Intel®Core m3-7Y30 at 1.61GHz with 8 GB RAM. For the first phase of the proposed approach, we used Spyder IDE with OWLready package. However, *Protégé* can also be used for manual extraction of terms (classes, properties, and description). For the rest of the phases, we used the R Project (R) for statistical computing. The operating system used is Windows 10. Subsequently, We used the ''''tm'', ''worldcloud'', ''snowballC'', ''xlsx'', ''clues'', ''factoextra'' and ''cluster'' R packages to perform the experiments. We adopted the best software engineering and programming standards [85] to implement the proposed system.

B. PSEUDOCODE FOR ORGANIZE ONTOLOGIES

Pseudocode 1 aims at describing two main activities. The first activity selects the best weighting method for un-supervised learners (USLs) used in this study. The second activity selects the best-unsupervised learner out of K-Means, K-Medoids-Euclidian, K-Medoids-Manhattan and Fuzzy c-means.

Pseudocode 1 Organise ontologies

```
Start
 Input:
  O: All groups of ontologies from the ontology
  repository.
  k: The number of clusters to organize the ontologies.
procedure:
 foreach (ontology o in ontology repository O) do
    ontology crawling and extract the terms.
    generate a text file $t_{(o)}$ containing terms for each
    ontology.
    Perform pre-processing activities and generate a
    VSM of t.
end
foreach (weighting method wm ) do
    apply wm to VSM.
end
foreach (unsupervised learner ul ) do
    apply ul technique to organize the ontologies O of a
    group into k clusters.
    evaluate the performance of ul with wm using
    evaluation criteria (average accuracy in
    Section VI(A)).
    Select the best wm and ul with the highest accuracy
    against each ontology group.
end
End
Result:
Best weighting method and corresponding best
un-supervised algorithm for organizing ontologies.
```

C. PSEUDOCODE 2: DETERMINE ONTOLOGY DOMAIN

Pseudocode 2 describes how to determine an appropriate ontology domain (for example, Computer Science, Academics) for any given UR.

D. PSEUDOCODE 3: SELECT THE MOST APPROPRIATE ONTOLOGY

The aim of Pseudocode 3 is to describe how to select the most appropriate ontology for the given UR from the ontology domain/category determined earlier by Pseudocode 2.

VIII. RESULTS AND DISCUSSION

This section discusses the results and findings of the proposed study. The efficiency of the proposed framework is evaluated in terms of organization of ontologies, predicting correct ontology group and recommendation of ontologies with respect to user requirements.

A. ORGANIZATION OF ONTOLOGIES

We assess the effectiveness of the proposed framework to organize ontologies into related groups (clusters) with respect to the expert's opinion. We took the help of domain experts to

Pseudocode 2 Determine ontology domain

```
Start
Input:
UR: A set of user requirements described in the context
of the target ontology group.
O: All groups of ontologies from the ontology
repository.
k: The number of clusters to organize the ontologies.
procedure:
foreach (user requirement ur in UR) and ontology o in
O do
    ontology crawling and extract the terms.
    generate a text file $t_{(o)}$ containing terms for each
    ontology.
    Perform pre-processing activities and generate a
    VSM of ur and t.
end
foreach (weighting method wm and unsupervised
learner ul) do
    apply wm to VSM.
    apply ul technique to organize the ontologies O and
    UR into k clusters.
    evaluate the performance of ul with a corresponding
    wm using evaluation criteria.
    Select the best wm and ul with the highest accuracy
    to determine the appropriate ontology group o for
    the selected ur.
end
End
Result:
The suggested ontology group.
```

Pseudocode 3 Select the most appropriate ontology

```
Start
Input:
Ontology_vectors:set of feature vectors of all the
ontologies in the recommended group against user
requirement.
Requirement_Vectors: Feature vector of given user
requirement.
Procedure:
foreach (Ontology_vector and Requirement_Vector ) do
    generate cosine value CS of Ontology_vector and
    Requirement_Vector.
    formulate a CS matrix containing Ontology_vectors
    and Requirement_Vector.
    while (Ontology_vector)
    Compare the CS of Requirement_Vector against CS
    of Ontology_vector.
    suggest the most appropriate ontology o having the
    highest CS value against Requirement_Vector.
    end while
end
End
Result:
The suggested ontology for the given user requirements.
```

identify the correct ontology group/category for each ontology, for example, if an ontology belongs to the networking, cybersecurity, or software systems category. We used these opinions as true labels and measured the accuracy of the proposed system for organizing the correct ontology group for each ontology. The experiments are performed according to the given procedure and results are reported with respect to the proposed evaluation model. In the context of the ontology organization, we used four algorithms and five weighting methods. The experimental results are shown in Fig 7. Furthermore, the results of organization of ontologies are also presented in Table 4,5,6, and 7. The key findings of the experimental results are summarized as follows:

- In the case of the Academics ontology domain, we find that the best weighting method for K-means is Entropy (with accuracy 0.55) whereas TFIDF is best for the other USLs. Similarly, in the case of the People ontology domain, we observe that Entropy (with accuracy 0.70) is the best weighting method for K-Medoids-Manhattan whereas TFIDF performs best for the other USLs. Moreover, in case of the Computer Science ontology domain, we observe that Binary and entropy (with accuracy 0.71) are the best weighting method for Fuzzy c-means and K-Medoids. Finally, in case of the Food and Drinks ontology domain, we observe that TFC (with accuracy 0.61) is the best weighting method for K-Medoids-Manhattan whereas Binary performs best for the other USLs.
- In case of the Academics ontology domain, we observe that Fuzzy c-means (Accuracy=0.61) outperforms the other USLs with their best weighting methods in terms of ontology organization.
- In case of the People ontology domain, we observe K-means (uracy=0.70), K-Medoids-Euclidean (Accuracy=0.70), K-Medoids-Manhattan (Accuracy=0.70) outperform Fuzzy C-means (Accuracy=0.68) with the best weighting methods in terms of ontology organization.
- In case of the Computer Science ontology domain, we observe Fuzzy c-means and K-Medoids (Accuracy=0.71) outperforms the rest of USLs with their best weighting methods in terms of ontology organization.
- In case of the Food and Drinks ontology domain, we observe Fuzzy c-means (Accuracy=0.61) and K-Medoids-Manhattan (Accuracy=0.61) outperform the rest of USLs with their best weighting methods in terms of ontology organization.
- We observe no single outperforming weighting method for USLs across all ontology domains. For

TABLE 4. **Academics Domain.**

	Binary	TFIDF	LTC	TFC	Entropy
Fuzzy c-means	0.55	0.61	0.52	0.55	0.52
K-Means	0.52	0.52	0.52	0.52	0.55
K-Medoids-Euclidian	0.52	0.52	0.52	0.52	0.52
K-Medoids-Manhattan	0.52	0.55	0.52	0.52	0.52

TABLE 5. **People Domain.**

	Binary	TFIDF	LTC	TFC	Entropy
Fuzzy c-means	0.51	0.66	0.59	0.59	0.62
K-Means	0.62	0.7	0.66	0.59	0.66
K-Medoids-Euclidian	0.51	0.7	0.66	0.59	0.62
K-Medoids-Manhattan	0.51	0.66	0.62	0.59	0.7

TABLE 6. **Computer Science Domain.**

	Binary	TFIDF	LTC	TFC	Entropy
Fuzzy c-means	0.71	0.7	0.62	0.67	0.71
K-Means	0.7	0.68	0.68	0.68	0.7
K-Medoids-Euclidian	0.71	0.64	0.64	0.64	0.71
K-Medoids-Manhattan	0.71	0.7	0.64	0.7	0.71

TABLE 7. **Food and Drinks Domain.**

	Binary	TFIDF	LTC	TFC	Entropy
Fuzzy c-means	0.61	0.52	0.52	0.52	0.57
K-Means	0.52	0.52	0.47	0.47	0.52
K-Medoids-Euclidian	0.57	0.52	0.52	0.52	0.52
K-Medoids-Manhattan	0.52	0.52	0.61	0.61	0.52

example, in case of th Academics and the People ontology domains, TFIDF performs best for Fuzzy c-means. However, Entropy and Binary work best for Fuzzy c-means for Computer Science, and Food and Drinks ontology domains, respectively.

- We also observe that an outperforming USL with its best weighting method cannot produce significant results across all domains. For example, Fuzzy c-means with TDIDF as best weighting method can outperform other USLs in case of the Academics domain (Accuracy=0.61) but cannot outperform other USLs for the People domain (Accuracy=0.68).

B. PREDICTING CORRECT ONTOLOGY GROUP FOR USER REQUIREMENT

We use the Pseudocode-2 and the proposed evaluation model to assess the effectiveness of the proposed framework in predicting the candidate ontology group for a given user requirement. The experimental results are shown in Fig 8. Furthermore, the results are also presented in tabular form in Table 8. In this section, we discuss the experimental results with respect to four ontology domains and 31 user requirements. In this regard, each USL is used with the best weighting method. The main findings of the experimental results terms of predicting the appropriate ontology category for the given UR are as follows.

- In case of the Academics ontology domains, we observe that Fuzzy c-means (Accuracy=0.61) outperforms the rest of USLs with their best weighting methods.

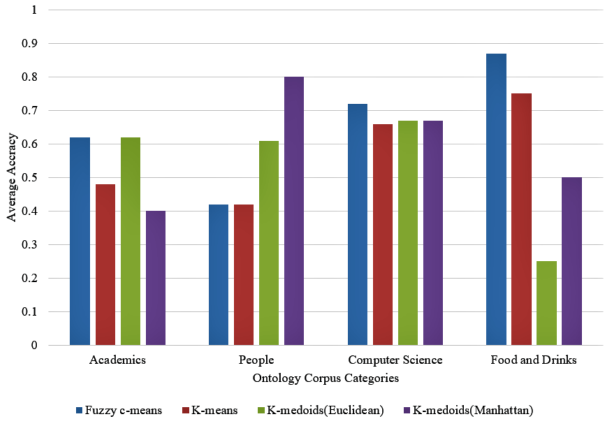


FIGURE 8. **The average accuracy of USLs with best weighting method for candidate ontology group determination.**

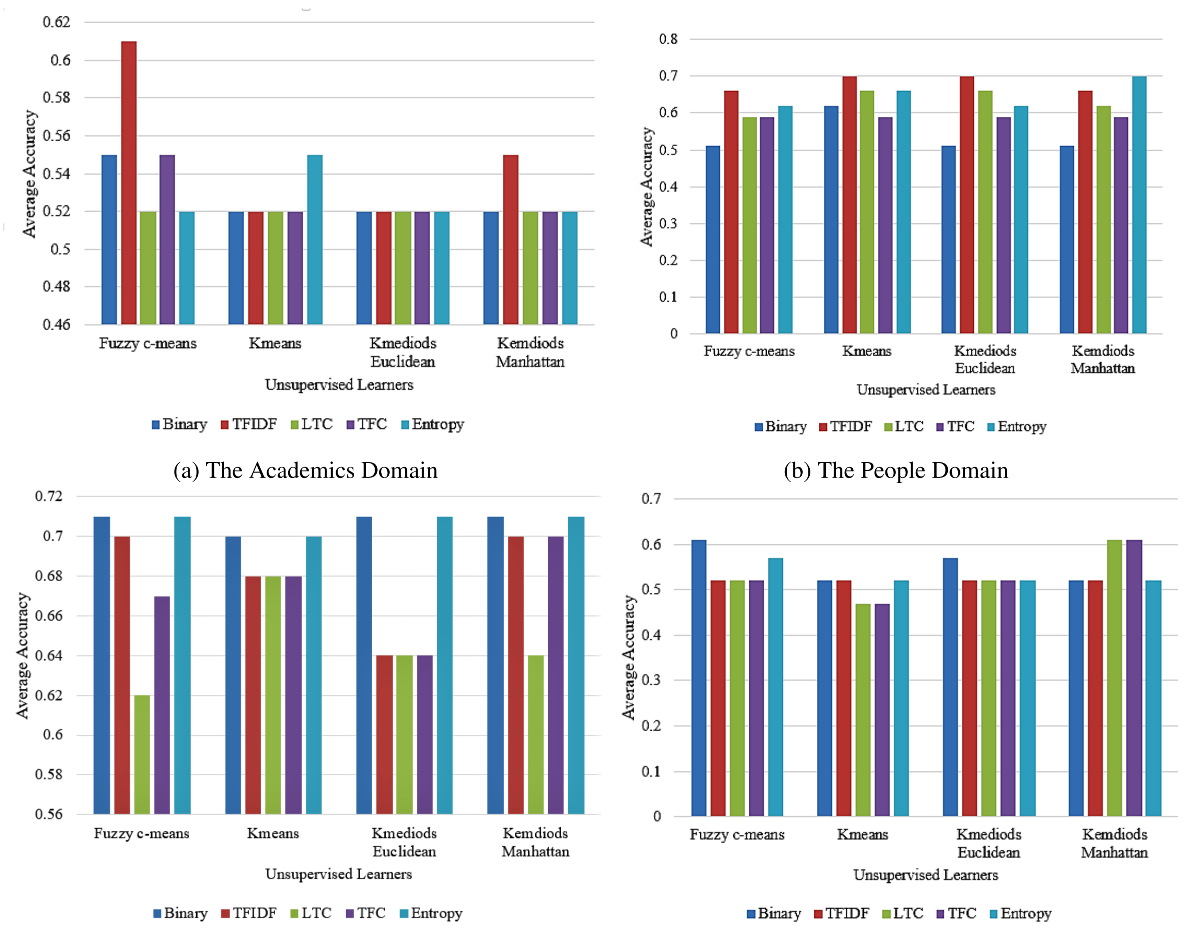
- In case of the People ontology domain, we observe that K-Medoid-Manhattan (Accuracy=0.80) outperforms K-Medoid-Euclidean (Accuracy=0.61), K-Means (Accuracy=0.42), and Fuzzy c-means (Accuracy=0.42) with their best weighting methods.
- In case of the Computer Science ontology domain, we observe that Fuzzy c-means (Accuracy=0.72) outperforms the rest of USLs with their best weighting methods.
- In the case of the Food and Drinks ontology domain, we observe Fuzzy c-means (Accuracy=0.88) outperform the rest of USLs with their best weighting methods.
- Finally, we observe that the performance of USLs varies with respect to the nature and size of the data.

C. ONTOLOGY RECOMMENDATION

We use the pseudocode-3 and the proposed evaluation model to assess the effectiveness of the proposed framework in selecting the correct ontology from the candidate ontology group for a given user requirement. The candidate group selected from the outperforming USL. In this regard, the cosine value of each ontology with respect to the user requirement is shown in Appendix A. The ontology with the highest cosine value is recommended as the right ontology for each UR. For example, in the case of UR-7 (Table 10 in Appendix A), the university information system ontology with the highest cosine value is recommended as the right ontology. The effectiveness of the proposed framework is evaluated for predicting correct ontology for the given UR in terms of RCO.

The key findings of the experiments results are summarized as follows:

- The proposed framework correctly recommended 8 out of 8 ontologies (RCO=100%) for the Academics domain, 5 out of 6 (RCO=83%) for the People group, 7 out of 9 (RCO=77%) ontologies for the Computer Science domain and 7 out of 8



(a) The Academics Domain

(b) The People Domain

(c) The Computer Science Domain

(d) Food and Drinks Domain

FIGURE 7. The accuracy of un-supervised learners with corresponding weighting method for the four ontology domains.

TABLE 8. Predicting Correct Ontology Group Against User Requirement.

	Fuzzy c-means	K-Means	K-Medoids-Euclidian	K-Medoids-Manhattan
Academics	0.62	0.48	0.62	0.4
People	0.42	0.42	0.61	0.8
Computer Science	0.72	0.66	0.67	0.67
Food and Drinks	0.87	0.75	0.25	0.5

ontologies (RCO=87%) for the Food and Drinks domain.

- It is observed that the proposed framework recommends 27 ontologies correctly for 31 URs, which describe the RCO as 87%.
- Moreover, it is also observed that the description of a UR plays a vital role in predicting an appropriate ontology for it.

Considering the promising results of the proposed system in the selection of appropriate ontology, the proposed system can be utilized to recommend ontologies to the users. The proposed system can help the novice or expert ontology designers, data providers, data, and knowledge engineers to accurately find the appropriate ontology. These data providers and engineers, are often overwhelmed by the search results or find it too time-consuming to find an already existing ontology because of time constraints. Subsequently, they end up creating one which is already available, doubling the cost. Moreover, unlike Linked Open Vocabularies [86], the only general ontology search engine [87] which provides popularity based ontologies in an unordered list, the proposed system recommends the only appropriate ontology to the user. The functionality can be enhanced to recommend the top three ontologies based on the cosine values.

IX. CONCLUSION AND FUTURE WORK

The aim of the proposed framework is to organize and recommend ontologies with respect to user requirements in order to reduce the efforts and time of developers. The proposed framework employs text categorization approach and un-supervised learning algorithms. The purpose of the proposed framework is to overcome the issue

TABLE 9. Computer science ontology recommendation for user requirements.

User Requirement	Sub-category	Expert Predicted Ontology	Ontologies in Group	CS Value
UR-1	Software Systems	Software Bugs Ontology	Computer Research Ontology	0.00
			Software Bugs Ontology	**0.61**
			Java Ontology	0.02
			DOGO Ontology	0.00
			Code Ontology	0.01
			ITSMO Ontology	0.13
			Software Package and Distribution Ontology	0.12
			SW-1 Ontology	0.07
			SW-2 Ontology	0.02
			E-commerce Software Ontology	0.00
			MUGshot Ontology	0.16
UR-2	Cybersecurity	STIX Exploit Ontology	Attack Pattern Ontology	0.19
			CVE Ontology	0.15
			CWE Ontology	0.13
			CYBOX Ontology	0.11
			Malware-1 Ontology	0.13
			Malware-2 Ontology	0.10
			STIX Exploit Ontology	**0.26**
			Unified Cybersecurity Ontology	0.17
UR-3	Networking	Network Server Ontology	IP Ontology	0.04
			TN Ontology	0.00
			WSDL Ontology	0.00
			DHCP Ontology	0.10
			DNS Ontology	0.18
			Server Ontology	**0.58**
			NIC Ontology	0.10
			Network System Ontology	0.05
			IP –Net Ontology	0.05
			Configurations Ontology	0.10
			JPA Ontology	0.07
			t-Chair Network Ontology	0.00
			Host Ontology	0.12
UR-4	Sentiments/ Emotions	Emotions Ontology	OntoSenticNet Ontology	0.14
			Emotions Ontology	**0.21**
			Adseek Emotions Ontology	0.16

TABLE 10. Academics ontology recommendation for user requirements.

User Requirement	Sub-category	Expert Predicted Ontology	Ontologies in Group	CS Value
UR-5	University/ Institute	University Benchmark Ontology	**University Benchmark Ontology**	**0.30**
			University Information System Ontology	0.11
			AISO University Ontology	0.06
			TMDU Ontology	0.05
			University Benchmark Ontology 2	0.12
UR-6	Research Publications	eBiquity Publication Ontology	Ekaw Ontology	0.00
			BibTex Ontology	0.00
			DPLB2 Ontology	0.00
			Confious Research Article Related Ontology	0.00
			eBiquity Publication Ontology	**0.16**
			ISWC Ontology	0.07
			PCS Ontology	0.00
			Onto Benchmark	0.00
UR-7	University/ Institute	University Information System Ontology	University Benchmark Ontology	0.16
			University Information System Ontology	**0.36**
			AISO University Ontology	0.05
			TMDU Ontology	0.05
			University Benchmark Ontology 2	0.14

TABLE 11. Food and Drinks ontology recommendation for user requirements.

User Requirement	Sub-category	Expert Predicted Ontology	Ontologies in Group	CS Value
UR-8	Drinks	Coffee Ontology	Bevon Ontology	0.09
			Beer Ontology	0.11
			Drink Ontology	0.12
			Wine-1 Ontology	0.02
			Wine-2 Ontology	0.00
			Whiskey Ontology	0.03
			Coffee Ontology	**0.33**
			DAML Wine ontology	0.09
			Cocktails Ontology	0.16
UR-9	Food	DC Pizza Ontology	ePizzza Ontology	0.18
			Pizza Ontology	0.18
			Pizza6 Ontology	0.13
			DC Pizza Ontology	**0.53**
			Food Ontology	0.17
			Food CR Ontology	0.08
			FoodON Ontology	0.00
			FoodON-1 Ontology	0.10
			FoodON Siren Ontology	0.11
			Food-1 Ontology	0.12
			Food-2 Ontology	0.09
UR-10	Drinks	Drinks Ontology	Bevon Ontology	0.11
			Beer Ontology	0.24
			Drink Ontology	**0.32**
			Wine-1 Ontology	0.02
			Wine-2 Ontology	0.00
			Whiskey Ontology	0.06
			Coffee Ontology	0.12
			DAML Wine ontology	0.02

of ontology selection in terms of their reusability. Moreover, we also proposed an evaluation model to assess the efficacy of the proposed framework. We evaluate the proposed framework in the context of four ontology domains with 31 URs.

The key implications of results of the proposed framework are as follows. Firstly, no single algorithm can be described as the best algorithm for the organization of ontologies and determination of correct ontology group for a given UR. Secondly, for determination of correct ontology group for a given UR, Fuzzy c-means performs best for the Academics domain whereas K-Medoids(Euclidian and Manhattan) performs better for the People and the Food and Drinks domains. Thirdly, it is observed that no single weighting method can be recommended as best for all USLs across all the four ontology domains. Fourthly, the proposed system recommends appropriate ontology to the user with RCO=87%. Fifthly, though the inclusion and exclusion of ontologies from the corpus might alter the presented results, it has no effect on the context of the proposed framework. This feature means that the proposed framework is not a context-aware system like existing approaches. Sixthly, like the existing approach for ontologies recommendation, the proposed framework does not need a formal specification of ontologies.

In the future, we will focus on two aspects: 1) to use n-gram for construction of feature vectors rather than the use of individual words, and 2) to assess the effectiveness of the proposed framework by considering numerous ontologies from different domains and more user requirements while focusing on the other multi-label text categorization approaches.

APPENDIX A
ONTOLOGY RECOMMENDATION RESULTS

This section presents the results of the ontology recommendation. Each requirement, candidate ontology group and CS value is presented in each table. Table 9 contains the Computer Science ontology group's results, Table 10 presents the results of the Academic ontology group. Similarly,

TABLE 12. People ontology recommendation for user requirements.

User Requirement	Sub-category	Expert Predicted Ontology	Ontologies in Group	CS Value
UR-11	Contact Information	Contact Ontology	**Phone Number Ontology**	**0.09**
			Address Ontology	0.03
			Contact Details Ontology	0.06
UR-12	Family	Obituary Ontology	**Obituary Ontology**	**0.34**
			SWRL Family Ontology	0.03
			Family Bond Ontology	0.00
			Family Tree Ontology	0.00
			People Ontology 2	0.00
			Family Relationship Ontology	0.05
			Drive Check Ontology	0.03
			People's Life Events Ontology	0.00
			TDWG Person LSID Ontology	0.11
UR-13	Work	Artist Work Ontology	Proton Ontology	0.00
			Artist Work Ontology	**0.34**
			Employees Mapped Ontology	0.00
			Task Ontology	0.00
			USA Army Ontology	0.00
			Production Role Ontology	0.00

Tables 11 and 12 present the results of Food and Drinks, and People's ontology groups respectively.

ACKNOWLEDGMENT

The authors are immensely grateful to the three domain experts who helped them in this research by identifying the ontologies against each requirements. They would also like to show our gratitude to the students of Semantic Web who provided them with the requirements for four ontology groups.

REFERENCES

[1] A. Lausch, A. Schmidt, and L. Tischendorf, "Data mining and linked open data – new perspectives for data analysis in environmental research," *Ecological Model.*, vol. 295, pp. 5–17, Jan. 2015.

[2] *The Exponential Growth of Data*. Accessed: May 3, 2020. [Online]. Available: https://insidebigdata.com/2017/02/16/the-exponential-growth-of-data/

[3] M. d'Aquin and N. F. Noy, "Where to publish and find ontologies? A survey of ontology libraries," *J. Web Semantics*, vol. 11, pp. 96–111, Mar. 2012.

[4] A. Abello, O. Romero, T. B. Pedersen, R. Berlanga, V. Nebot, M. J. Aramburu, and A. Simitsis, "Using semantic Web technologies for exploratory OLAP: A survey," *IEEE Trans. Knowl. Data Eng.*, vol. 27, no. 2, pp. 571–588, Feb. 2015.

[5] Y. Ding, "Semantic Web: Who is who in the field—A bibliometric analysis," *J. Inf. Sci.*, vol. 36, no. 3, pp. 335–356, Jun. 2010.

[6] Y. K. Hooi, M. F. Hassan, and A. M. Shariff, "Ontology evaluation—A criteria selection framework," in *Proc. Int. Symp. Math. Sci. Comput. Res.*, 2015, pp. 298–303.

[7] B. Dutta, "Examining the interrelatedness between ontologies and linked data," *Library Hi Tech*, vol. 35, no. 2, pp. 312–331, Jun. 2017.

[8] M. Devi and M. Dua, "ADANS: An agriculture domain question answering system using ontologies," in *Proc. Int. Conf. Comput., Commun. Autom. (ICCCA)*, May 2017, pp. 122–127.

[9] H. Zheng, Y. Wang, C. Han, F. Le, R. He, and J. Lu, "Learning and applying ontology for machine learning in cyber attack detection," in *Proc. 17th IEEE Int. Conf. Trust, Secur. Privacy Comput. Commun./12th IEEE Int. Conf. Big Data Sci. Eng. (TrustCom/BigDataSE)*, Aug. 2018, pp. 1309–1315.

[10] C. Ochs, Y. Perl, J. Geller, S. Arabandi, T. Tudorache, and M. A. Musen, "An empirical analysis of ontology reuse in BioPortal," *J. Biomed. Informat.*, vol. 71, pp. 165–177, Jul. 2017.

[11] M. Talebpour, M. Sykora, and T. Jackson, "The role of community and social metrics in ontology evaluation: An interview study of ontology reuse," in *Proc. 9th Int. Joint Conf. Knowl. Discovery, Knowl. Eng. Knowl. Manage.*, 2017, pp. 119–127.

[12] A. S. Butt, A. Haller, and L. Xie, "Ontology search: An empirical evaluation," in *Proc. Int. Semantic Web Conf.*, 2014, pp. 130–147.

[13] Y. Djenouri, A. Belhadi, and R. Belkebir, "Bees swarm optimization guided by data mining techniques for document information retrieval," *Expert Syst. Appl.*, vol. 94, pp. 126–136, Mar. 2018.

[14] S. Hussain, J. Keung, M. K. Sohail, A. A. Khan, and M. Ilahi, "Automated framework for classification and selection of software design patterns," *Appl. Soft Comput.*, vol. 75, pp. 1–20, Feb. 2019.

[15] R. K. Saha, L. Zhang, S. Khurshid, and D. E. Perry, "An information retrieval approach for regression test prioritization based on program changes," in *Proc. IEEE/ACM 37th IEEE Int. Conf. Softw. Eng.*, vol. 1, May 2015, pp. 268–279.

[16] H. Alani, N. F. Noy, N. Shah, N. Shadbolt, and M. A. Musen, "Searching ontologies based on content: Experiments in the biomedical domain," in *Proc. 4th Int. Conf. Knowl. Capture (K-CAP)*, 2007, pp. 55–62.

[17] C. Jonquet, M. A. Musen, and N. H. Shah, "Building a biomedical ontology recommender Web service," *J. Biomed. Semantics*, vol. 1, no. 1, p. S1, Jun. 2010.

[18] M. Martínez-Romero, J. M. Vázquez-Naya, J. Pereira, and A. Pazos, "BiOSS: A system for biomedical ontology selection," *Comput. Methods Programs Biomed.*, vol. 114, no. 1, pp. 125–140, Apr. 2014.

[19] A. Groza, I. Dragoste, I. Sincai, I. Jimborean, and V. Moraru, "An ontology selection and ranking system based on the analytic hierarchy process," in *Proc. 16th Int. Symp. Symbolic Numeric Algorithms Sci. Comput.*, Sep. 2014, pp. 293–300.

[20] A. S. Butt, A. Haller, and L. Xie, "RecOn: Ontology recommendation for structureless queries," *Appl. Ontology*, vol. 11, no. 4, pp. 301–324, Feb. 2017.

[21] N. Trokanas and F. Cecelja, "Ontology evaluation for reuse in the domain of process systems engineering," *Comput. Chem. Eng.*, vol. 85, pp. 177–187, Feb. 2016.

[22] J. Aguilar, J. Altamiranda, and O. Portilla, "Hybrid recommender system of biomedical ontologies," in *Proc. Latin Amer. Comput. Conf. (CLEI)*, Oct. 2016, pp. 1–12.

[23] R. B. K. Brown, G. Beydoun, G. Low, W. Tibben, R. Zamani, F. García-Sánchez, and R. Martinez-Bejar, "Computationally efficient ontology selection in software requirement planning," *Inf. Syst. Frontiers*, vol. 18, no. 2, pp. 349–358, Apr. 2016.

[24] N. Z. Zulkarnain, F. Meziane, and G. Crofts, "A methodology for biomedical ontology reuse," in *Natural Language Processing and Information Systems* (Lecture Notes in Computer Science), vol. 9612. Cham, Switzerland: Springer, 2016, pp. 3–14.

[25] M. Martínez-Romero, C. Jonquet, M. J. O'Connor, J. Graybeal, A. Pazos, and M. A. Musen, "NCBO ontology recommender 2.0: An enhanced approach for biomedical ontology recommendation," *J. Biomed. Semantics*, vol. 8, no. 1, pp. 1–22, Dec. 2017.

[26] E. Faessler, F. Klan, A. Algergawy, B. König-Ries, and U. Hahn, "Selecting and tailoring ontologies with JOYCE," in *Knowledge Engineering and Knowledge Management* (Lecture Notes in Computer Science), vol. 10180. Cham, Switzerland: Springer, 2017, pp. 114–118.

[27] S. Hussain, J. Keung, A. A. Khan, and K. E. Bennin, "A methodology to automate the selection of design patterns," in *Proc. IEEE 40th Annu. Comput. Softw. Appl. Conf. (COMPSAC)*, vol. 2, Jun. 2016, pp. 161–166.

[28] S. Hussain, J. Keung, and A. A. Khan, "Software design patterns classification and selection using text categorization approach," *Appl. Soft Comput.*, vol. 58, pp. 225–244, Sep. 2017.

[29] S. Hussain, J. Keung, A. A. Khan, A. Ahmad, S. Cuomo, F. Piccialli, G. Jeon, and A. Akhunzada, "Implications of deep learning for the automation of design patterns organization," *J. Parallel Distrib. Comput.*, vol. 117, pp. 256–266, Jul. 2018.

[30] I. Androutsopoulos, G. Paliouras and E. Michelakis, "Learning to filter unsolicited commercial E-mail," Athens Univ. Econ. Bus., Athens, Greece, Nat. Centre Sci. Res., Paris, France, Tech. Rep. 2004/2, Mar. 2004.

[31] D. Vilares, C. Gómez-Rodríguez, and M. A. Alonso, "Universal, unsupervised (rule-based), uncovered sentiment analysis," *Knowl.-Based Syst.*, vol. 118, pp. 45–55, Feb. 2017.

[32] D. Kılıç, A. Özçift, F. Bozyigit, P. Yıldırım, F. Yücalar, and E. Borandag, "TTC-3600: A new benchmark dataset for turkish text categorization," *J. Inf. Sci.*, vol. 43, no. 2, pp. 174–185, Apr. 2017.

[33] F. A. Zaghoul and S. Al-Dhaheri, "Arabic text classification based on features reduction using artificial neural networks," in *Proc. UKSim 15th Int. Conf. Comput. Modeling Simulation*, Apr. 2013, pp. 485–490.

[34] M. Karan, J. Snajder, and B. D. Basic, "Evaluation of classification algorithms and features for collocation extraction in croatian," in *Proc. 8th Int. Conf. Lang. Resource Eval.*, no. 1, 2012, pp. 657–662.

[35] A. Basile, G. Dwyer, M. Medvedeva, J. Rawee, H. Haagsma, and M. Nissim, "N-GrAM: New Groningen author-profiling model," 2017, *arXiv:1707.03764*. [Online]. Available: https://arxiv.org/abs/1707.03764

[36] J. Y. Liu, "A survey of deep learning approaches for recommendation systems," *J. Phys., Conf. Ser.*, vol. 1087, 2018, Art. no. 062022.

[37] Y. Li, S. Wang, Q. Pan, H. Peng, T. Yang, and E. Cambria, "Learning binary codes with neural collaborative filtering for efficient recommendation systems," *Knowl.-Based Syst.*, vol. 172, pp. 64–75, May 2019.

[38] D. Lian, R. Liu, Y. Ge, K. Zheng, X. Xie, and L. Cao, "Discrete content-aware matrix factorization," in *Proc. 23rd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2017, pp. 325–334.

[39] W.-C. Kang and J. McAuley, "Candidate generation with binary codes for large-scale top-N recommendation," in *Proc. 28th ACM Int. Conf. Inf. Knowl. Manage.*, Nov. 2019, pp. 1523–1532.

[40] K. Zhou and H. Zha, "Learning binary codes for collaborative filtering," in *Proc. 18th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD)*, 2012, pp. 498–506.

[41] H. Liu, X. He, F. Feng, L. Nie, R. Liu, and H. Zhang, "Discrete factorization machines for fast feature-based recommendation," in *Proc. Int. Joint Conf. Artif. Intell. (IJCAI)*, Jul. 2018, pp. 3449–3455.

[42] X. Liu, J. He, C. Deng, and B. Lang, "Collaborative hashing," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 2147–2154.

[43] Z. Lu, Y. Hu, Y. Jiang, Y. Chen, and B. Zeng, "Learning binary code for personalized fashion recommendation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 10554–10562.

[44] C.-W. Chen, S.-P. Tseng, T.-W. Kuan, and J.-F. Wang, "Outpatient text classification using attention-based bidirectional LSTM for robot-assisted servicing in hospital," *Information*, vol. 11, no. 2, p. 106, Feb. 2020.

[45] J. P. A. Vieira and R. S. Moura, "An analysis of convolutional neural networks for sentence classification," in *Proc. Latin Amer. Comput. Conf. (CLEI)*, Sep. 2017, pp. 1–5.

[46] H. Zhang, L. Xiao, Y. Wang, and Y. Jin, "A generalized recurrent neural architecture for text classification with multi-task learning," in *Proc. 26th Int. Joint Conf. Artif. Intell.*, Aug. 2017, pp. 2873–2879.

[47] Y. Luan and S. Lin, "Research on text classification based on CNN and LSTM," in *Proc. IEEE Int. Conf. Artif. Intell. Comput. Appl. (ICAICA)*, Mar. 2019, pp. 352–355.

[48] F. Ali, S. El-Sappagh, S. M. R. Islam, A. Ali, M. Attique, M. Imran, and K.-S. Kwak, "An intelligent healthcare monitoring framework using wearable sensors and social networking data," *Future Gener. Comput. Syst.*, vol. 114, pp. 23–43, Jan. 2021.

[49] F. Ali, S. El-Sappagh, S. M. R. Islam, D. Kwak, A. Ali, M. Imran, and K.-S. Kwak, "A smart healthcare monitoring system for heart disease prediction based on ensemble deep learning and feature fusion," *Inf. Fusion*, vol. 63, pp. 208–222, Nov. 2020.

[50] G. Chen, D. Ye, Z. Xing, J. Chen, and E. Cambria, "Ensemble application of convolutional and recurrent neural networks for multi-label text categorization," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, May 2017, pp. 2377–2383.

[51] A. Farquhar, R. Fikes, and J. Rice, "The ontolingua server: A tool for collaborative ontology construction," *Int. J. Hum.-Comput. Stud.*, vol. 46, no. 6, pp. 707–727, Jun. 1997.

[52] C. Ramesh and K. V. C. Rao, "Ontology based Web usage mining model," in *Proc. Int. Conf. Inventive Commun. Comput. Technol. (ICICCT)*, 2017, pp. 356–362.

[53] F. Ali, P. Khan, K. Riaz, D. Kwak, T. Abuhmed, D. Park, and K. S. Kwak, "A fuzzy ontology and SVM–Based Web content classification system," *IEEE Access*, vol. 5, pp. 25781–25797, 2017.

[54] B. Swartout, R. Patil, K. Knight, and T. Russ, "Toward distributed use of large-scale ontologies," in *Proc. 10th Knowl. Acquisition Workshop (KAW)*, Banff, AB, Canada, Nov. 1996.

[55] A. A. Alsanad, A. Chikh, and A. Mirza, "A domain ontology for software requirements change management in global software development environment," *IEEE Access*, vol. 7, pp. 49352–49361, 2019.

[56] M. A. Medina-Nieto, "An overview of ontologies," Center Res. Inf. Automat. Technol., Interact. Cooperat. Technol. Lab., Univ. de las Américas Puebla, Puebla, Mexico, Tech. Rep., 2003.

[57] S. El-Sappagh, J. M. Alonso, F. Ali, A. Ali, J.-H. Jang, and K.-S. Kwak, "An ontology-based interpretable fuzzy decision support system for diabetes diagnosis," *IEEE Access*, vol. 6, pp. 37371–37394, 2018.

[58] Y. Ma, P. Zhang, and J. Ma, "An ontology driven knowledge block summarization approach for chinese judgment document classification," *IEEE Access*, vol. 6, pp. 71327–71338, 2018.

[59] M. E. Ibrahim, Y. Yang, D. L. Ndzi, G. Yang, and M. Al-Maliki, "Ontology-based personalized course recommendation framework," *IEEE Access*, vol. 7, pp. 5180–5199, 2019.

[60] F. Alsubaei, A. Abuhussein, and S. Shiva, "Ontology-based security recommendation for the Internet of medical things," *IEEE Access*, vol. 7, pp. 48948–48960, 2019.

[61] D. Cavaliere, V. Loia, and S. Senatore, "Towards an ontology design pattern for UAV video content analysis," *IEEE Access*, vol. 7, pp. 105342–105353, 2019.

[62] S. Siddiqui, M. A. Rehman, S. Muhammad Doudpota, and A. Waqas, "Ontology driven feature engineering for opinion mining," *IEEE Access*, vol. 7, pp. 67392–67401, 2019.

[63] A. Qazi and R. H. Goudar, "An ontology-based term weighting technique for Web document categorization," *Procedia Comput. Sci.*, vol. 133, pp. 75–81, Jan. 2018.

[64] M. Krendzelak and F. Jakab, "Text categorization with machine learning and hierarchical structures," in *Proc. 13th Int. Conf. Emerg. eLearning Technol. Appl. (ICETA)*, Nov. 2015, pp. 1–5.

[65] A. K. Uysal and S. Gunal, "The impact of preprocessing on text classification," *Inf. Process. Manage.*, vol. 50, no. 1, pp. 104–112, Jan. 2014.

[66] A. Hotho, A. Hotho, A. Nürnberger, and G. Paaß, "A brief survey of text mining," in *Proc. LDV FORUM-Gld. J. Comput. Linguist. Lang. Technol.*, 2005.

[67] T. Korenius, J. Laurikkala, K. Järvelin, and M. Juhola, "Stemming and lemmatization in the clustering of finnish text documents," in *Proc. 13th ACM Conf. Inf. Knowl. Manage. (CIKM)*, 2004, p. 625.

[68] M. Allahyari *et al.*, "A brief survey of text mining: Classification, clustering and extraction techniques," Jul. 2017, *arXiv:1707.02919*. [Online]. Available: https://arxiv.org/abs/1707.02919

[69] M. Lan, C. Lim Tan, J. Su, and Y. Lu, "Supervised and traditional term weighting methods for automatic text categorization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 4, pp. 721–735, Apr. 2009.

[70] G. Salton and C. Buckley, "Term-weighting approaches in automatic text retrieval," *Inf. Process. Manage.*, vol. 24, no. 5, pp. 513–523, Jan. 1988.

[71] T. Wang, Y. Cai, H.-F. Leung, Z. Cai, and H. Min, "Entropy-based term weighting schemes for text categorization in VSM," in *Proc. IEEE 27th Int. Conf. Tools with Artif. Intell. (ICTAI)*, Nov. 2015, pp. 325–332.

[72] C. Zhang, X. Wu, Z. Niu, and W. Ding, "Authorship identification from unstructured texts," *Knowl.-Based Syst.*, vol. 66, pp. 99–111, Aug. 2014.

[73] E. Saraç and S. A. Özel, "An ant colony optimization based feature selection for Web page classification," *Sci. World J.*, vol. 2014, pp. 1–16, 2014.

[74] I. Idris and A. Selamat, "Improved email spam detection model with negative selection algorithm and particle swarm optimization," *Appl. Soft Comput.*, vol. 22, pp. 11–27, Sep. 2014.

[75] W. Medhat, A. Hassan, and H. Korashy, "Sentiment analysis algorithms and applications: A survey," *Ain Shams Eng. J.*, vol. 5, no. 4, pp. 1093–1113, Dec. 2014.

[76] D. Fensel, *Ontologies: A Silver Bullet for Knowledge Management and Electronic Systems*. Berlin, Germany: Springer, 2004.

[77] S. Fraihat, "Ontology-concepts weighting for enhanced semantic classification of documents," *Int. J. Innov. Comput. Inf. Control*, vol. 12, no. 2, pp. 519–531, 2016.

[78] D. A. Simovici and C. Djeraba, *Clustering*, No. 9781447164067, 2014.

[79] S. Al-Anazi, H. AlMahmoud, and I. Al-Turaiki, "Finding similar documents using different clustering techniques," *Procedia Comput. Sci.*, vol. 82, pp. 28–34, Mar. 2016.

[80] P. Arora, Deepali, and S. Varshney, "Analysis of K-means and K-medoids algorithm for big data," *Procedia Comput. Sci.*, vol. 78, pp. 507–512, Dec. 2016.

[81] A. Saxena, M. Prasad, A. Gupta, N. Bharill, O. P. Patel, A. Tiwari, M. J. Er, W. Ding, and C.-T. Lin, "A review of clustering techniques and developments," *Neurocomputing*, vol. 267, pp. 664–681, Dec. 2017.

[82] R. Subhashini and V. J. S. Kumar, "Evaluating the performance of similarity measures used in document clustering and information retrieval," in *Proc. 1st Int. Conf. Integr. Intell. Comput.*, Aug. 2010, pp. 27–31.

[83] R. Rifkin and A. Klautau, "In defense of one-vs-all classification," *J. Mach. Learn. Res.*, vol. 5, pp. 101–141, Dec. 2004.

[84] M. Dragoni, S. Poria, and E. Cambria, "OntoSenticNet: A commonsense ontology for sentiment analysis," *IEEE Intell. Syst.*, vol. 33, no. 3, pp. 77–85, May 2018.

[85] M. Javed, B. Ahmad, S. Hussain, and S. Ahmad, "Mapping the best practices of XP and project management: Well defined approach for project manager," *J. Comput.*, vol. 2, no. 3, pp. 2151–9617, 2010.

[86] P.-Y. Vandenbussche, G. A. Atemezing, M. Poveda-Villalón, and B. Vatant, "Linked open vocabularies (LOV): A gateway to reusable semantic vocabularies on the Web," *Semantic Web*, vol. 8, no. 3, pp. 437–452, Dec. 2016.

[87] A. S. Butt, "Ontology search: Finding the right ontologies on the Web," in *Proc. Int. World Wide Web Conf.*, Aug. 2015, pp. 487–491.

MUHAMMAD AZEEM SARWAR received the bachelor's degree in software engineering from NUML Islamabad. He is currently pursuing the degree with the Department of Computer Science, COMSATS University Islamabad, Islamabad, Pakistan. His research interests include software engineering, smart health, big data analytics, semantic web, and machine learning.

MANSOOR AHMED received the Ph.D. degree from the Vienna University of Technology, Vienna, Austria. He is currently working as an Assistant Professor with the Department of Computer Science, COMSATS University Islamabad, Islamabad, Pakistan. His research interests include information security and privacy, distributed computing, knowledge-based systems, data provenance, and semantic web technologies.

ASAD HABIB received the Doctor of Engineering degree from the Graduate School of Information Science, Nara Institute of Science and Technology (NAIST), Japan. He is currently the Director of the Institute of Information Technology (IIT), Kohat University of Science and Technology, Kohat, Pakistan. His research interests include data science, natural language processing, computational modeling, software engineering, knowledge-based organizational analytics, prediction, and recommender systems.

MUHAMMAD KHALID received the M.S. degree in computer science from the Institute of Management Sciences, Peshawar, Pakistan. He is currently pursuing the Ph.D. degree with Northumbria University, Newcastle Upon Tyne. He is also working as a Research Associate with University of Lincoln, U.K. His research interests include machine learning, human–robot interaction, EV charging and scheduling, the Internet of Things, wireless sensor networks, and autonomous valet parking.

M. AKHTAR ALI received the B.Sc., M.Sc., and Ph.D. degrees. He worked on a two-year aKTP project at the Corbridge Medical Group to develop a new general practice service delivery model informed by data mining to predict population need and consequently redesigned processes and workforce focus. He is currently a Senior Lecturer with the Department of Computer and Information Sciences, Northumbria University. He has extensive experience of research and development in the field of database systems, data integration, database migration, query optimization, data warehousing, materialized views, and data mining. He worked on a KTP project in the area of geographical information systems. He has extensive experience of supervising B.Sc., M.Sc., and Ph.D. students/projects in computing and information sciences. He is leading new postgraduate program development in data science.

MOHSIN RAZA (Member, IEEE) received the B.S. (Hons.) and M.S. degrees in electronic engineering from Mohammad Ali Jinnah University (MAJU), Pakistan, and the Ph.D. degree from the Math, Physics and Electrical Engineering Department, Northumbria University (NU), U.K. He worked as a Lecturer at Northumbria University, U.K., from 2019 to 2020; a Postdoctoral Fellow with Middlesex University, U.K., from 2018 to 2019; a Demonstrator/Associate-Lecturer and a Doctoral Fellow at NU, from 2015 to 2017; a Junior Lecturer and a Lecturer with the Engineering Department, Mohammad Ali Jinnah University, Pakistan, from 2010 to 2012 and from 2012 to 2015, respectively; and a Hardware Support Engineer at Unified Secure Services, Pakistan, from 2009 to 2010. He is currently a Senior Lecturer with the Department of Computer Science, Edge Hill University, U.K. His research interests include the IoT, 5G and wireless networks, autonomous transportation systems, machine learning, Industry 4.0, and digital twins. He served as a Technical Committee Member for ICET 2012, SKIMA 2015, SKIMA 2017, WSGT 2017, CSNDSP 2018, SKIMA 2018, ICT 2019, and CSoNet 2019. He has been a Guest Editor for *International Journal of Distributed Sensor Networks*, Special Issue on Heterogenous Internet of Medical Things, and a Reviewer for several journals, including IEEE ACCESS, IEEE COMMUNICATIONS LETTERS, *Sensors* (MDPI), *Vehicular Communications* (Elsevier), and the *Arabian Journal for Science and Engineering* (Springer).

SHAHID HUSSAIN has worked as an Assistant Professor with the Department of Computer Science, COMSATS University Islamabad, Islamabad, Pakistan. He is currently working as a Pro Tem Faculty Member with the Department of Computer and Information Science, University of Oregon (UO), Oregon, USA. He is also a member with the High Performance Computing (HPC) Laboratory, UO. His research interests include empirical software engineering, software design patterns, text mining, fault prediction, and machine learning.

GHUFRAN AHMED received the Ph.D. degree from the Department of Computer Science, Mohammad Ali Jinnah University (renamed to Capital University of Science and Technology), Islamabad, in 2013, and the Ph.D. degree from the Faculty of Computer Science and Engineering, GIK Institute, Topi. He completed the postdoctoral training at the Department of Computer Science and Digital Technology, Faculty of Engineering and Environment, Northumbria University, Newcastle Upon Tyne, U.K., in 2016. He was an Assistant Professor with the Department of Computer Science, COMSATS University Islamabad, Pakistan. He has been an Associate Professor with the Department of Computer Science, FAST NUCES, since January 2020. His research interests include the IoT, wireless sensor networks, and wireless body area networks. He also worked as a Visiting Scholar with the CReWMaN Lab, Department of Computer Science and Engineering, The University of Texas at Arlington, from 2008 to 2009. He has served as a Guest Editor for the special sections of *International Journal of Distributed Sensor Networks* (IJDSN), *Journal of Sensors* (Hindawi), and *Journal of Wireless Communication and Mobile Computing* (Hindawi). He is also working as an Associate Editor of IEEE ACCESS, an Academic Editor for *Wireless Communications and Mobile Computing* (Hindawi), and *Journal of Sensors* (Hindawi). He is also serving as an Editorial Board Member for *Ad Hoc & Sensor Wireless Networks* (AHSWN).

• • •