International Journal of Innovative
Computing, Information and Control
Volume **14**, Number **3**, June **2018**

 ISSN 1349-4198
pp. **817–832**

A MODIFIED KNOWLEDGE DISCOVERY PROCESS IN THE TEXT DOCUMENTS

Rajni Jindal and Shweta

Computer Engineering Department
Delhi Technological University
Shahbad Daulatpur, Main Bawana Road, New Delhi 110042, India
rajnijindal@dce.ac.in; shweta.madhur21@gmail.com

Received September 2017; revised January 2018

Abstract. *To discover useful information or knowledge from text documents, the knowledge discovery in text documents (KDT) process is used. In this paper, an automated knowledge discovery process is proposed that helps in the multi label categorization of text documents, i.e., a text document may belong to more than one class or category. We have proposed a modified lexical-semantics based knowledge discovery process for text documents (LS-KDT). The proposed process consists of seven phases. These are – text document collection, data preprocessing, lexical analysis/scanner, semantic analysis, classification, ranking of labels and knowledge discovery. The proposed process is implemented on two text datasets. The first dataset consists of the research articles of computer science domain. The research articles are randomly selected from ACM digital library. The second dataset contains research articles of medical domain. These articles are also taken from ACM digital library. To test the performance, standard performance measures like recall, precision and F-measure are calculated. The performance of the proposed process is compared with the results of standard taxonomy used by ACM digital library and our proposed process shows a significantly better performance.*
Keywords: Knowledge discovery in text documents (KDT) process, Lexical analysis, Semantic analysis, Multi label learning

1. **Introduction.** The process of data mining or knowledge discovery in databases (KDD) deals with the extraction of useful information or knowledge from huge amounts of data [1,2]. It is used in structured databases. The field of text mining has emerged from data mining. It performs mining on unstructured text documents. The text documents are news stories, emails, research papers, reports, contracts, etc. The knowledge discovery in text databases (KDT) process is a variant of KDD process used in text databases [4].

The technique of text categorization is an important research area. It categorizes a text document into a specific category. The text categorization can be single label (also called as binary) or multi label in nature [3]. In our research, we have focused on multi label categorization of text documents. After categorization, a text document is ranked. The term ranking is the arrangement or ordering of class labels according to their relevance in multi label environment [6]. We have used this concept to find out the membership value of class labels and ordered them.

In our previous work, we had proposed a framework on categorization of text documents [5]. The framework consisted of lexical phase, semantic phase and classification phase. In this paper, we have extended the framework and proposed a modified KDT process known as LS-KDT for knowledge discovery in text documents. The proposed process performs automated categorization of text documents. The whole process is divided into a series

of sub processes called as phases. There are seven phases in our proposed LS-KDT process. These are – text document collection, data preprocessing, lexical analysis/scanner, semantic analysis, classification, ranking of labels and knowledge discovery. These are discussed in detail in Section 3.

In literature, authors have contributed in the field of knowledge discovery using different methods. They have used the concept of ontology [7], entity and relationships [9], etc. on text documents. However, nobody has focused on the categorization of research articles. We have taken research articles of computer science domain and medical domain and used lexical and semantics concepts to discover knowledge.

The main contribution of our proposed work is that it will help the research community to identify the exact category or categories to which a text document belongs. We have tested our proposed process on research articles. The proposed LS-KDT process works for both single label as well as multi label categorization. It will help in efficient searching and indexing of the research articles. The accurate categorization of articles also helps digital libraries, databases, repositories or online resources to efficiently store or search the articles.

The paper is organized as follows. Section 2 discusses the related work done in KDT process. In Section 3, we present our proposed LS-KDT process for knowledge discovery in text documents. The proposed process consists of seven phases with their description. Section 4 shows the details of experiments conducted with the results obtained. In this section, we give the details of datasets used and show the working of our proposed process on a sample research article. In Section 5, we have given the performance comparison of our proposed process with ACM digital library results. This is followed by conclusion in the next section.

2. **Related Work.** In today's world, there has been a lot of research done in the area of knowledge discovery. In this section, we present the related work done in knowledge discovery in different domains.

In automotive domain, an ontology based knowledge discovery system was proposed [7]. The textual reports were taken and faults were identified. In another work, knowledge discovery was done in inspection reports of marine structures [8]. The authors used concept extraction and linkage approach along with self-organizing map (SOM) for document organization. This aided in reporting the kinds of defects in the reports. In biomedical domain, the authors in [9] had developed a text mining system that focused on extraction of entities and relations. The system was tested on five corpora and had shown good results. Another work was done on biomedical documents [10]. A set of tools, Med-TAKMI was developed for medical documents. It is an extension of the TAKMI (Text Analysis and Knowledge MIning) system originally developed for text mining in customer relationship-management applications. It used keyword based search for extracting entities and parsers to find relations among entities. In the field of legal documents, author in [11] proposed a knowledge model. This model included collection of legal documents, preprocessed them and then grouped them using clustering technique of data mining. In [12], authors proposed a new method to identify criminal networks from a collection of text documents. Then useful information was extracted from them for investigation. Also, the method identified relationships between the criminals in a community.

Many authors have used semantic concepts in the area of text categorization. In [19], authors have combined semantic web concepts with regular expression for information retrieval from the web. In our work, we have used hypernym relationship of Word net to study the semantic relationship between tokens. In this way, we have identified similar

tokens by finding out tokens that have a common hypernym. This has helped to reduce the dimensionality.

To the best of our knowledge, the above authors have not focused on categorization of research articles using lexical and semantics concepts. We have extracted tokens from the articles, and then used Word net to identify semantic relationships between the tokens. In our work, a novel process of knowledge discovery based on lexical and semantics concepts is proposed which helps in categorization of research articles. Among the seven phases of the proposed LS-KDT process, three phases can contribute more to the literature. These are: lexical analysis/scanner, semantic analysis and ranking of labels. The first phase lexical analysis/scanner is based on the idea of extracting tokens and storing them with their frequency. The next phase is semantic analysis. It is based on the concept of using hypernym relationship of Word net. Then hypernym trees are drawn of all the tokens and common hypernym is identified. In this way, semantic relationship between tokens is used for dimension reduction.

3. **Proposed Lexical-Semantics Based Knowledge Discovery Process for Text Documents (LS-KDT).** The proposed process (LS-KDT) is subdivided into a series of steps called phases. There are seven phases in the process. These are: text document collection, data preprocessing, lexical analysis/scanner, semantic analysis, classification, ranking of labels and knowledge discovery. Figure 1 given shows the proposed LS-KDT process. The details of phases are given below.

Phase 1: Text Document Collection

In the first phase, a data warehouse is built from text documents. The text documents may be research articles, medical documents, legal documents, etc. The documents are multi label in nature. That is, a single text document may belong to a number of classes or categories.

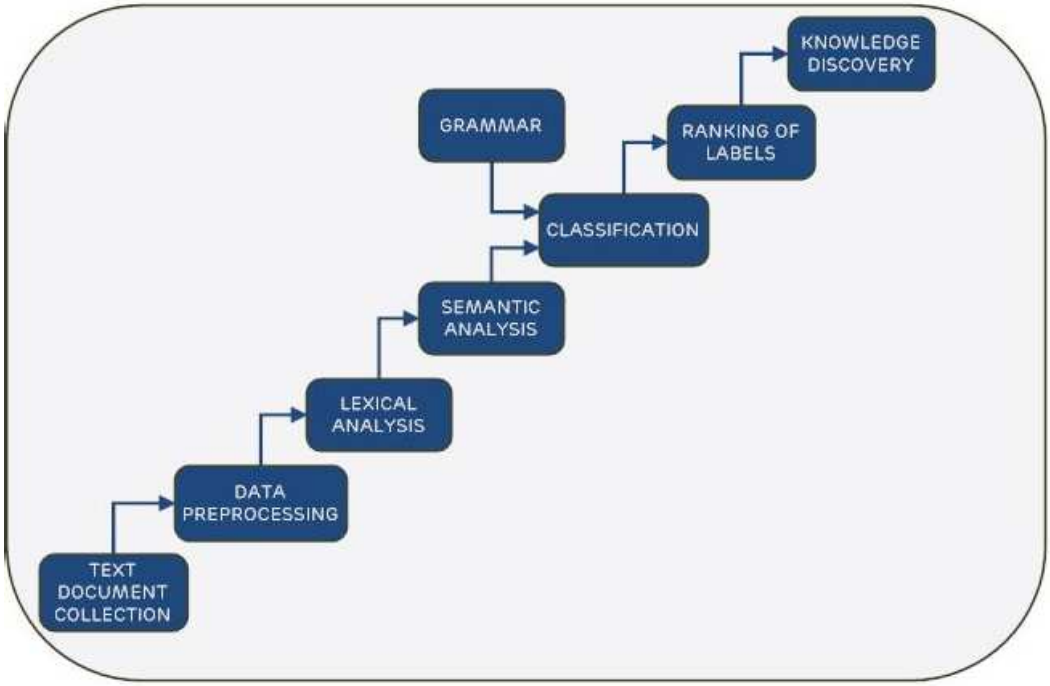


FIGURE 1. Architecture of the proposed LS-KDT process

Phase 2: Data Preprocessing

This is the second phase of the proposed LS-KDT process. In this phase, text document is prepared for mining by performing operations like data cleaning. This is an important phase as quality of results depends on the quality of data. If data is dirty then it will affect the mining results. Stop words are removed in this phase by using a predefined stop words list [20].

Phase 3: Lexical Analysis/Scanner

The lexical analysis phase or scanner scans the text document and identifies tokens. We have taken the datasets of research articles of computer science domain and medical domain. In this phase, title, abstract and keywords of a research article are taken as input.

The ACM computing classification system is a standard system (revised in 2012), hierarchical in nature [21]. It consists of broad categories or areas, which are further organized into sub categories. We have used the standard ACM computing classification system to identify the tokens. This phase identifies the tokens from the title, keywords and abstract of the article and stores them along with their frequency (of occurrence) in a table. The output of this phase is a list of tokens (t_i) along with their frequency of occurrence (f_{i1}) for each research article. That is, if a research article is J_i, then it can be represented as $\{(t_i, f_{i1}), (t_{i+1}, f_{i2}), (t_{i+2}, f_{i3}), \ldots\}$, where i varies from 1 to n (total number of tokens). The flow diagram of lexical analysis phase is given in Figure 2.

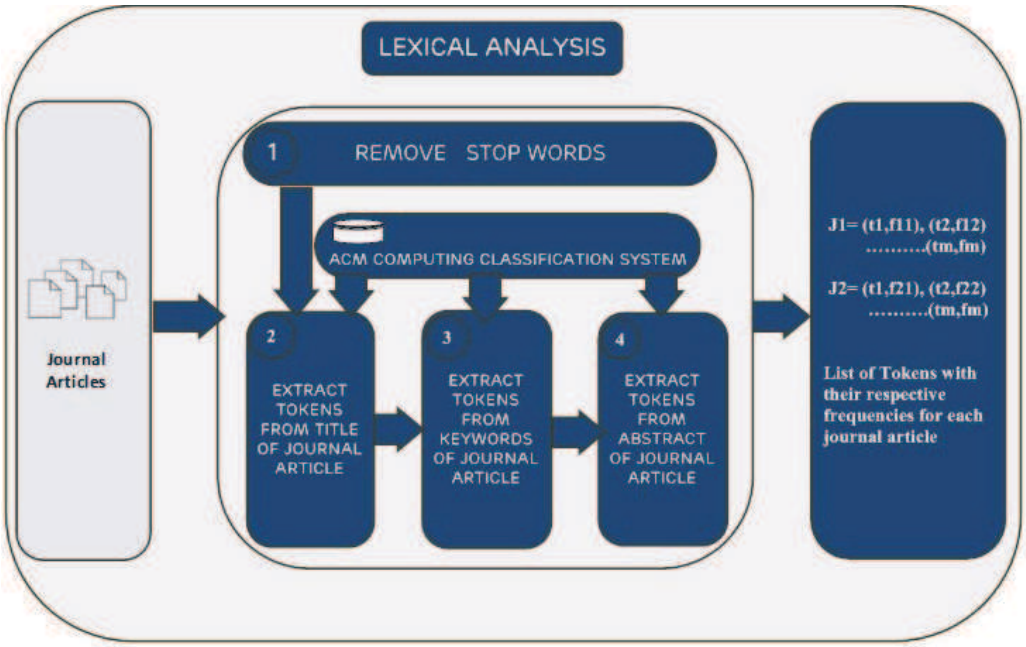


FIGURE 2. System flow of lexical analysis phase

Phase 4: Semantic Analysis

The next phase is semantic analysis. This phase receives a stream of tokens with their frequency as input from lexical analysis phase. The idea of semantics is used for dimensionality reduction, that is, to reduce the number of tokens. Firstly, the average frequency of tokens is calculated. The tokens are partitioned, i.e., all the tokens that have frequency greater than or equal to the average frequency are important tokens, and they are kept as singleton sets. The rest tokens are kept in the same set. Further, semantic relationships are analyzed between the tokens that are kept in the same set using Word net. Word net is a lexical database which stores semantic relationships of words [14]. The

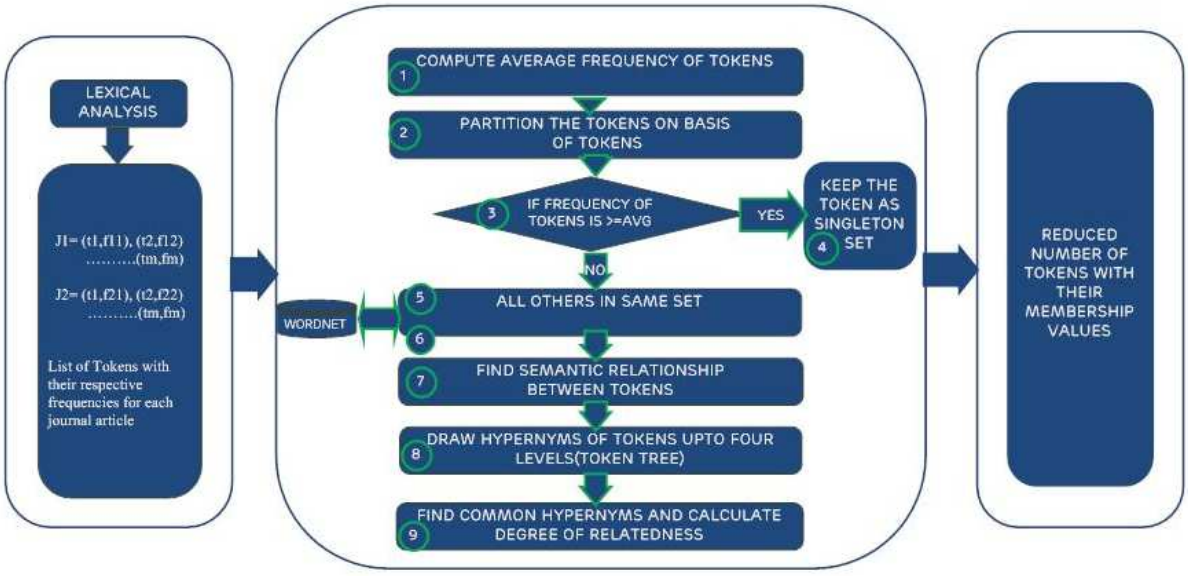


FIGURE 3. System flow of semantic analysis phase

1. IF (common hypernym of two tokens are at the same level) or (difference of levels is 0) THEN degree of relatedness is VERY HIGH, $d = 7$.
2. IF (common hypernym of two tokens are at a difference of level 1) THEN degree of relatedness is ALMOST HIGH, $d = 6$.
3. IF (common hypernym of two tokens are at a difference of level 2) THEN degree of relatedness is HIGH, $d = 5$.
4. IF (common hypernym of two tokens are at a difference of level 3) THEN degree of relatedness is LOW, $d = 4$.
5. IF (common hypernym of two tokens are at a difference of level 4) THEN degree of relatedness is VERY LOW, $d = 3$.

FIGURE 4. Rules to calculate degree of relatedness

different relationships are synonyms, hypernyms, hyponyms, meronyms, etc. The system flow of semantic analysis phase is given in Figure 3. We have used hypernym relationship of tokens [15]. The hypernyms of tokens are drawn up to four levels and tokens having common hypernym are identified.

If two tokens have a common hypernym, that means they are related. This relationship is measured by calculating the degree of relatedness between tokens. The degree of relatedness is calculated by using set of rules given in Figure 4.

To explain the above method, suppose t_1 and t_2 are two tokens that have a common hypernym. The membership value of a token (denoted by x) is calculated by the following formula:

$$x = \text{average}(f_1, f_2) + d,$$

where f_1 and f_2 are frequencies of the two tokens t_1 and t_2 respectively and d is degree of relatedness between them.

Out of the two tokens t_1 and t_2, one representative is selected and assigned the membership value x. If the token is a keyword, it is given preference; otherwise if it has greater frequency, it is preferred. This process is repeated. Finally, we get a reduced number of tokens with their membership values as output of this phase.

Phase 5: Classification

The next phase is classification or multi label classification. This phase receives input as tokens with their membership values from semantic analysis phase. It performs two functions. Firstly, it merges tokens (single) using the given keyword list of the article and a grammar. So, multi word tokens (collocations) are handled. (Till now, tokens were single words). The membership values of individual tokens are summed up. Secondly, this phase identifies classes of tokens using the standard ACM computing classification system. The output of this phase is a list of tokens (merged or collocated) with their membership values(x) and classes. The details of this phase are given in [5].

A grammar is constructed in this phase using the standard ACM computing classification system. The aim is to identify related tokens or collocations. A snapshot of grammar generated is shown below in Figure 5, where C is the set of all broad categories or areas of computer science domain.

1. C → Computer System Organization | Networks | Software and its Engineering | Theory of Computation | Mathematics of Computing | Information Systems | Security and Privacy | Human Centered Computing | Computing Methods | Applied Computing | Social and Professional Topics
2. Computer System Organization → Architecture | Embedded and cyber-physical systems | Real-time systems | Dependable and fault-tolerant systems and networks
3. Architecture → Serial Architecture | Parallel architectures | Distributed architectures | Other architectures
4. Serial Architecture → · · · · · ·
5. Parallel architectures → · · · · · ·

and so on

FIGURE 5. Snapshot of grammar generated

Phase 6: Ranking of Labels

The next phase is ranking of class labels. This phase receives a list of tokens with their membership values and classes. In this phase, we have proposed eight quantifiers – none, almost none, very low, low, high, higher, highest and all [6]. These quantifiers help in ordering of class labels in multi label categorization of text documents. The output of this phase is a ranking or ordering of class labels on the basis of membership values of tokens.

Phase 7: Knowledge Discovery

This is the last phase of the proposed LS-KDT process. The knowledge obtained is an ordering of membership values of class labels of a text document. This is a novel idea. For example, if we take a research article belonging to computer science discipline, the article may belong to any of the various sub disciplines under the computer science domain. This is a case of multi label categorization [17,18]. Using the concept of quantifiers here, we are able to calculate the membership degree of various class labels in a single research article. It helps in automatic categorization of articles. It will further help the editors to find the best reviewers or experts for the research articles.

4. **Experiments Conducted and Results Obtained.** We have conducted experiments on 128-bit machines with clock speed of 2.6 GHz. For implementation purpose, we have used Java 1.7 [16]. The details of datasets used are given in Section 4.1. In Section

TABLE 1. Details of dataset of computer science research articles

S. No	Category	Number of Articles
1	Computer System Organization	20
2	Networks	30
3	Software and Engineering	20
4	Theory of Computation	20
5	Mathematics of Computing	20
6	Information Systems	30
7	Security and Privacy	30
8	Human Centered Computing	20
9	Computing Methodologies	20
10	Applied Computing	20
11	Social and Professional Topics	20

TABLE 2. Details of dataset of medical domain articles

S. No	Category	Number of Articles
1	Computer System Organization	20
2	Networks	20
3	Software and Engineering	20
4	Theory of Computation	20
5	Mathematics of Computing	25
6	Information Systems	20
7	Security and Privacy	30
8	Human Centered Computing	30
9	Computing Methodologies	30
10	Applied Computing	30
11	Social and Professional Topics	30

4.2, we explain the working of the proposed process with the help of a sample research article.

4.1. **Dataset details.** We have conducted experiments on two datasets. One is a dataset of research articles of computer science domain. The articles are randomly selected from ACM digital library. We have used a subset of taxonomy of ACM digital library. There are 11 categories under computer science domain in ACM taxonomy that we have used. In total, we have considered 250 articles. The details of the dataset are given in Table 1. The second dataset contains articles from medical domain. The articles are randomly selected from ACM digital library. We have taken a total of 275 articles. Table 2 shows the details of the second dataset.

4.2. **Results on a sample research article.** To explain the working of the proposed LS-KDT process, the results are shown for a sample research article given in Figure 6. The title of the sample article is "Mining Community Structures in Multidimensional Networks".

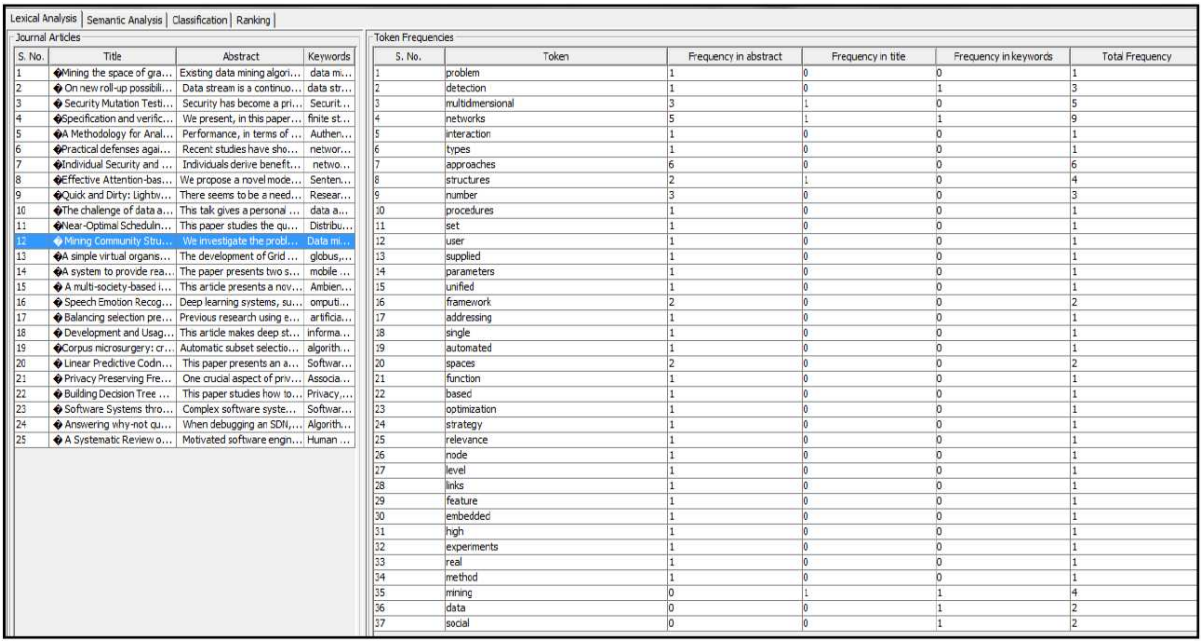
The snapshots obtained after the execution of our proposed LS-KDT process are given below. Figure 7 shows the output of lexical analysis phase. It displays the list of total 37 tokens along with their frequency identified in the title, abstract and keywords of the article. This list is fed as input to semantic analysis module. Figure 8 shows the final

Title: "Mining Community Structures in Multidimensional Networks".

Abstract: We investigate the problem of community detection in multidimensional networks, that is, networks where entities engage in various interaction types (dimensions) simultaneously. While some approaches have been proposed to identify community structures in multidimensional networks, there are a number of problems still to solve. In fact, the majority of the proposed approaches suffer from one or even more of the following limitations: (1) difficulty detecting communities in networks characterized by the presence of many irrelevant dimensions, (2) lack of systematic procedures to explicitly identify the relevant dimensions of each community, and (3) dependence on a set of user-supplied parameters, including the number of communities, that require a proper tuning. Most of the existing approaches are inadequate for dealing with these three issues in a unified framework. In this paper, we develop a novel approach that is capable of addressing the aforementioned limitations in a single framework. The proposed approach allows automated identification of communities and their sub-dimensional spaces using a novel objective function and a constrained label propagation-based optimization strategy. By leveraging the relevance of dimensions at the node level, the strategy aims to maximize the number of relevant within-community links while keeping track of the most relevant dimensions. A notable feature of the proposed approach is that it is able to automatically identify low dimensional community structures embedded in a high dimensional space. Experiments on synthetic and real multidimensional networks illustrate the suitability of the new method.

Keywords: Data mining, social networks, community detection

FIGURE 6. A sample research article



Lexical Analysis | Semantic Analysis | Classification | Ranking

Journal Articles

S. No.	Title	Abstract	Keywords
1	Mining the space of gra...	Existing data mining algori...	data mi...
2	On new roll-up possibili...	Data stream is a continuo...	data str...
3	Security Mutation Testi...	Security has become a pri...	Securit...
4	Specification and verific...	We present, in this paper...	finite st...
5	A Methodology for Anal...	Performance, in terms of ...	Authen...
6	Practical defenses agai...	Recent studies have sho...	networ...
7	Individual Security and ...	Individuals derive benefit...	netwo...
8	Effective Attention-bas...	We propose a novel mode...	Senten...
9	Quick and Dirty: Lightw...	There seems to be a need...	Resear...
10	The challenge of data a...	This talk gives a personal ...	data a...
11	Near-Optimal Scheduln...	This paper studies the qu...	Distribu...
12	Mining Community Stru...	We investigate the probl...	Data mi...
13	A simple virtual organs...	The development of Grid ...	globus,...
14	A system to provide rea...	The paper presents two s...	mobile ...
15	A multi-society-based i...	This article presents a nov...	Ambien...
16	Speech Emotion Recog...	Deep learning systems, su...	omputi...
17	Balancing selection pre...	Previous research using e...	artificia...
18	Development and Usag...	This article makes deep st...	informa...
19	Corpus microsurgery: cr...	Automatic subset selectio...	algorith...
20	Linear Predictive Codn...	This paper presents an a...	Softwar...
21	Privacy Preserving Fre...	One crucial aspect of priv...	Associa...
22	Building Decision Tree ...	This paper studies how to...	Privacy,...
23	Software Systems thro...	Complex software syste...	Softwar...
24	Answering why-not qu...	When debugging an SDN,...	Algorith...
25	A Systematic Review o...	Motivated software engn...	Human ...

Token Frequencies

S. No.	Token	Frequency in abstract	Frequency in title	Frequency in keywords	Total Frequency
1	problem	1	0	0	1
2	detection	1	0	1	3
3	multidimensional	3	1	0	5
4	networks	5	1	1	9
5	interaction	1	0	0	1
6	types	1	0	0	1
7	approaches	6	0	0	6
8	structures	2	1	0	4
9	number	3	0	0	3
10	procedures	1	0	0	1
11	set	1	0	0	1
12	user	1	0	0	1
13	supplied	1	0	0	1
14	parameters	1	0	0	1
15	unified	1	0	0	1
16	framework	2	0	0	2
17	addressing	1	0	0	1
18	single	1	0	0	1
19	automated	1	0	0	1
20	spaces	2	0	0	2
21	function	1	0	0	1
22	based	1	0	0	1
23	optimization	1	0	0	1
24	strategy	1	0	0	1
25	relevance	1	0	0	1
26	node	1	0	0	1
27	level	1	0	0	1
28	links	1	0	0	1
29	feature	1	0	0	1
30	embedded	1	0	0	1
31	high	1	0	0	1
32	experiments	1	0	0	1
33	real	1	0	0	1
34	method	1	0	0	1
35	mining	0	1	1	4
36	data	0	0	1	2
37	social	0	0	1	2

FIGURE 7. Output of lexical analysis phase

output of semantic analysis phase displaying the partitions made and reduced number of tokens (8 in number). Next the output of classification phase is shown in Figure 9. It shows the tokens along with the classes (their hierarchy) to which they belong and their

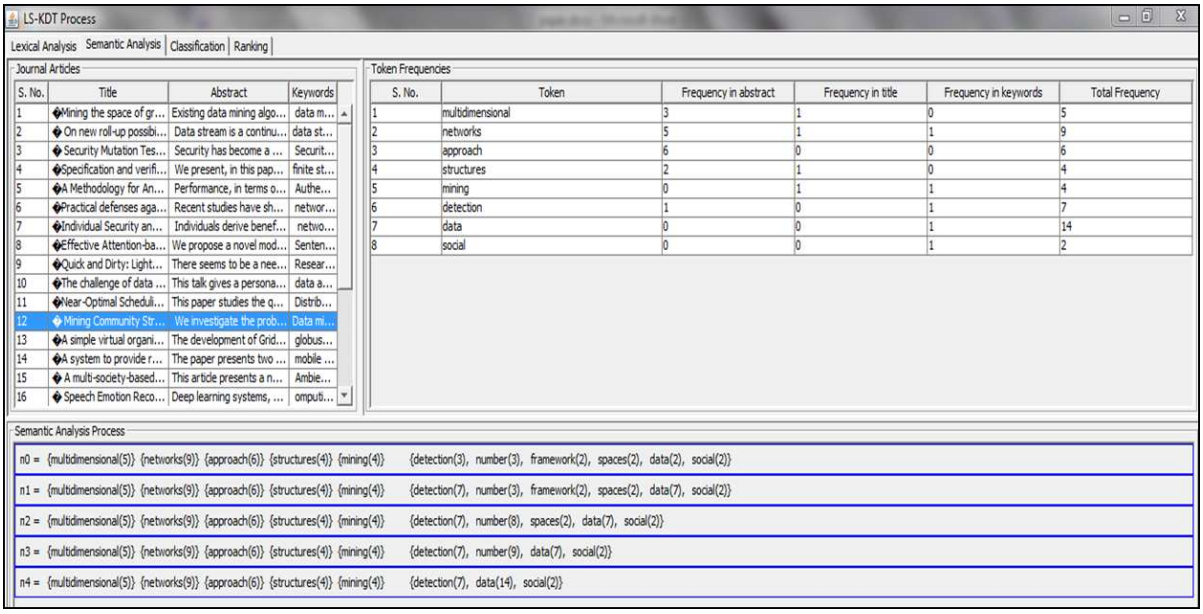
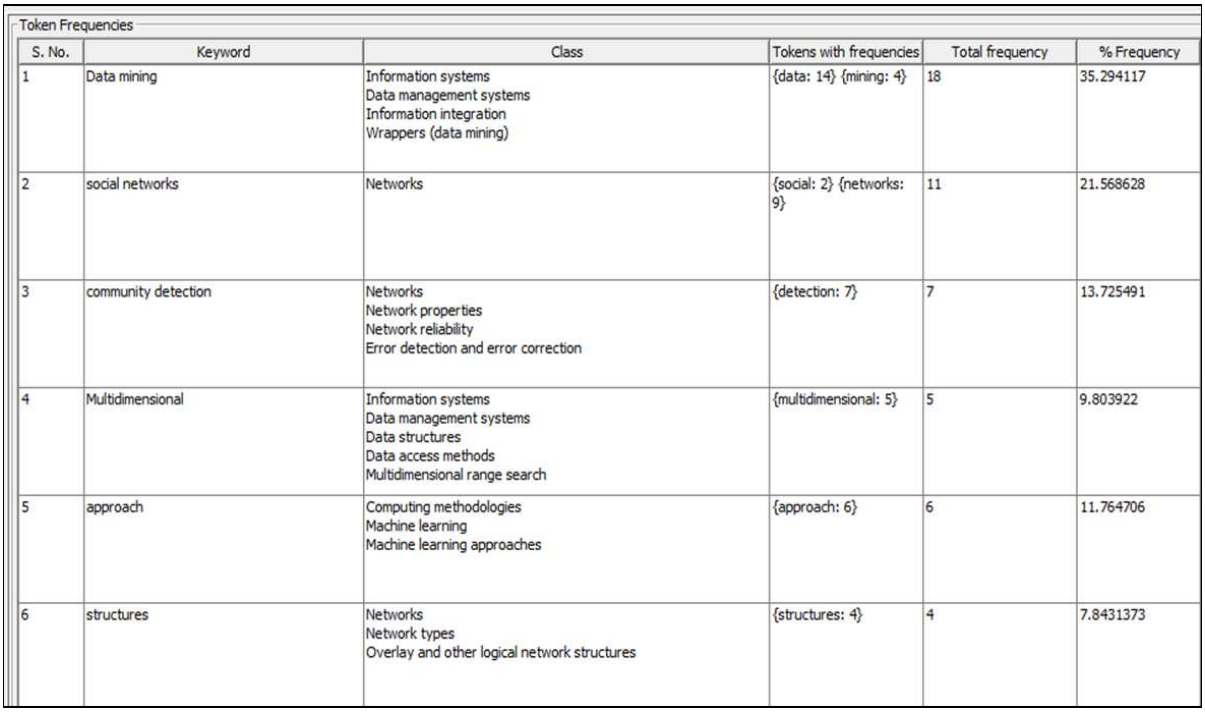


FIGURE 8. Final output of semantic analysis phase



Token Frequencies

S. No.	Keyword	Class	Tokens with frequencies	Total frequency	% Frequency
1	Data mining	Information systems Data management systems Information integration Wrappers (data mining)	{data: 14} {mining: 4}	18	35.294117
2	social networks	Networks	{social: 2} {networks: 9}	11	21.568628
3	community detection	Networks Network properties Network reliability Error detection and error correction	{detection: 7}	7	13.725491
4	Multidimensional	Information systems Data management systems Data structures Data access methods Multidimensional range search	{multidimensional: 5}	5	9.803922
5	approach	Computing methodologies Machine learning Machine learning approaches	{approach: 6}	6	11.764706
6	structures	Networks Network types Overlay and other logical network structures	{structures: 4}	4	7.8431373

FIGURE 9. Output of classification phase

membership values. Finally, Figure 10 shows the ordering of membership values of classes of tokens.

5. **Performance Comparison.** The performance of the proposed LS-KDT process is compared with the results of ACM digital library. ACM digital library uses CCS tool for the categorization of research articles. The CCS tool displays the result in a hierarchical manner, i.e., broad category of an article followed by levels of categories to which an

Token Frequencies

S. No.	Keyword	Class	Tokens with frequencies	Total frequency	% Frequency
1	structures	Networks Network types Overlay and other logical network structures	{structures: 4}	4	7.8431373
2	Multidimensional	Information systems Data management systems Data structures Data access methods Multidimensional range search	{multidimensional: 5}	5	9.803922
3	approach	Computing methodologies Machine learning Machine learning approaches	{approach: 6}	6	11.764706
4	community detection	Networks Network properties Network reliability Error detection and error correction	{detection: 7}	7	13.725491
5	social networks	Networks	{social: 2} {networks: 9}	11	21.568628
6	Data mining	Information systems Data management systems Information integration Wrappers (data mining)	{data: 14} {mining: 4}	18	35.294117

FIGURE 10. Output of ranking phase

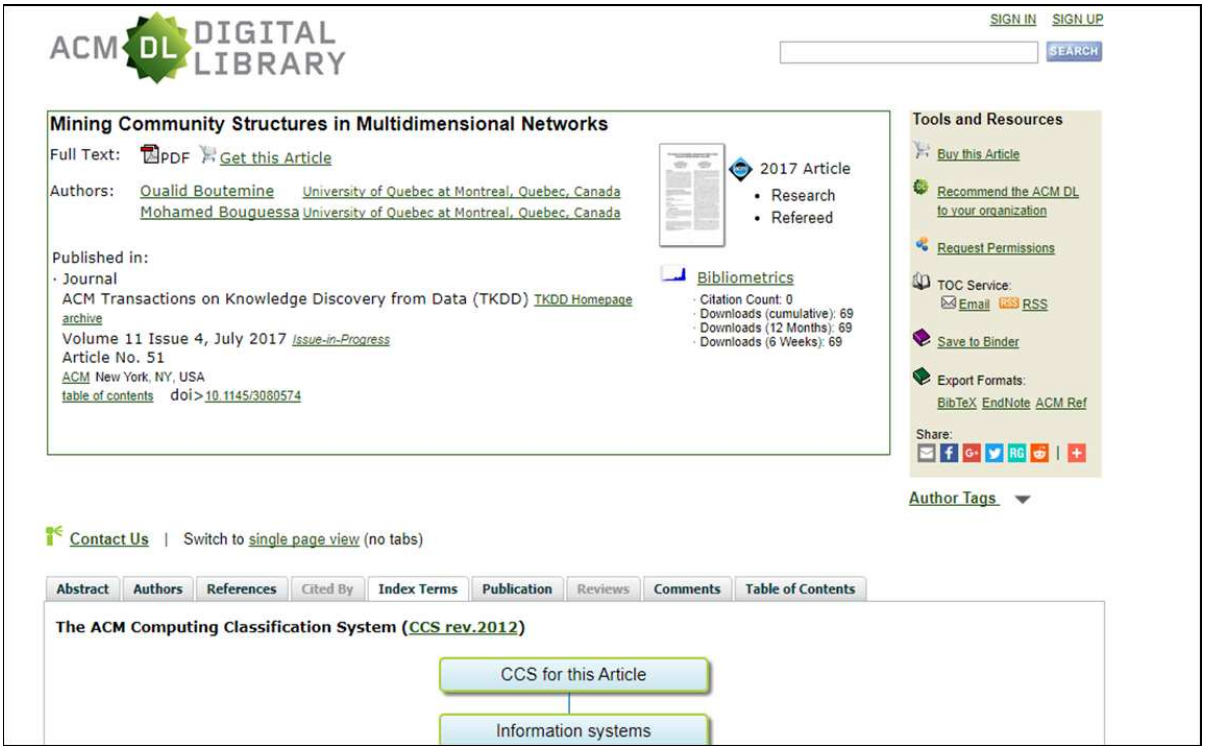


FIGURE 11. Sample article in ACM digital library

article belongs. Our proposed process also shows the levels of categories to which an article belongs.

The results shown by our proposed process are compared with the results of ACM digital library. And standard performance metrics like recall, precision and F-measure are calculated [13]. In our work, recall and precision can be calculated as shown in

Equations (1) and (2). F-measure is harmonic mean between precision and recall.

$$\text{Recall} = \frac{\sum \text{Relevant Levels}}{\sum \text{All the Levels within the Category}} \tag{1}$$

$$\text{Precision} = \frac{\sum \text{Relevant Levels}}{\sum \text{Retrieved Levels}} \tag{2}$$

$$\text{F-measure} = \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \tag{3}$$

To explain the calculation of recall and precision, the sample article in ACM digital library (that was shown above) is shown in Figure 11. The results of CCS tool used by ACM digital library are shown in Figure 12. The sample article belongs to one class, i.e., information systems. The results given by our proposed process on the same article are shown in Figure 10. It shows that the article belongs to information systems, networks and computing methodologies classes. Therefore, our proposed process has displayed all the classes to which a research article belongs. It has given a better categorization result.

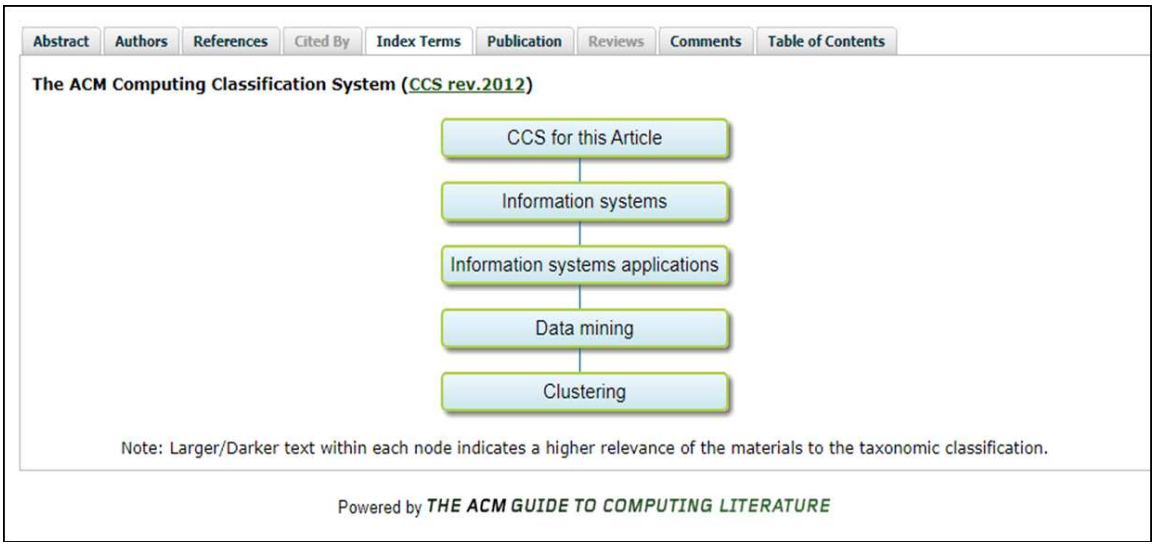


FIGURE 12. Results of sample article in ACM digital library

TABLE 3. Comparison of results for computer science domain articles

		ACM Digital Library			Proposed LS-KDT Process		
S. No	**Class/Category**	**Precision**	**Recall**	**F-Measure**	**Precision**	**Recall**	**F-Measure**
1	Computer System Organization	60	60	60	90	80	85
2	Networks	100	35	51	100	54	70
3	Software and Engineering	92	75	82	100	100	100
4	Theory of Computation	100	100	100	90	90	90
5	Mathematics of Computing	100	100	100	100	100	100
6	Information Systems	84	51	63	87	88	88
7	Security and Privacy	100	40	57	100	67	80
8	Human Centered Computing	90	60	72	95	75	84
9	Computing Methodologies	100	70	73	90	80	85
10	Applied Computing	94	55	69	95	67	79
11	Social and Professional Topics	85	50	63	90	55	68
	Average Values	Micro Precision = 91	Micro Recall = 63	Micro F-Measure = 72	Micro Precision = 94	Micro Recall = 78	Micro F-Measure = 85

5.1. **Comparison of results for computer science domain articles.** Table 3 shows the values of precision, recall and F-measure as computed from ACM digital library and the proposed LS-KDT process. These results are obtained for articles of computer science domain.

First we calculated the performance metrics of the categories one by one. Then, for the average values, we calculated micro precision, micro recall and micro F-measure. As shown in the above table, the values of micro precision, micro recall and micro F-measure

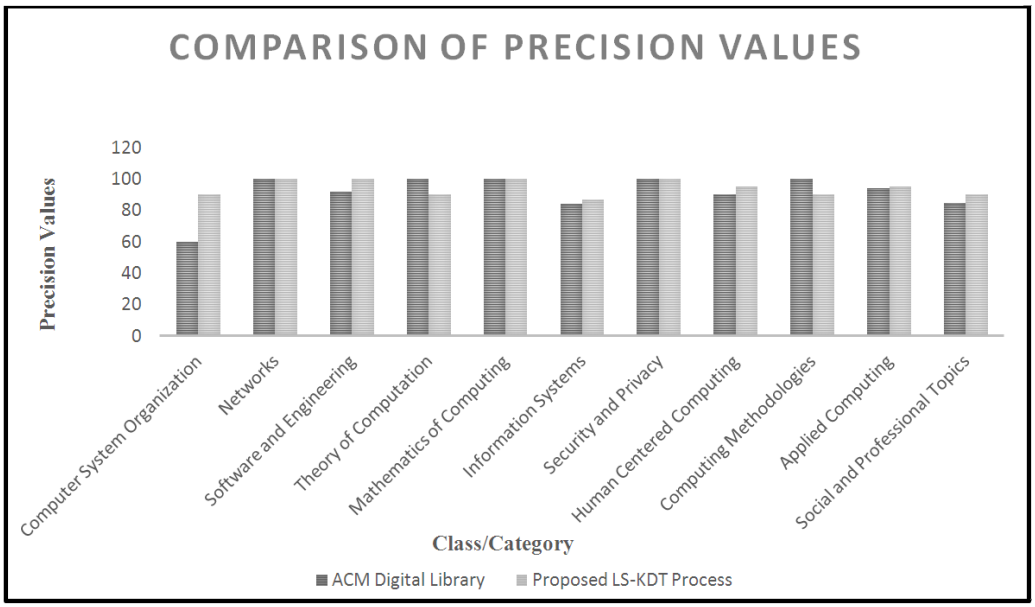


FIGURE 13. Comparison of precision values on computer science articles dataset

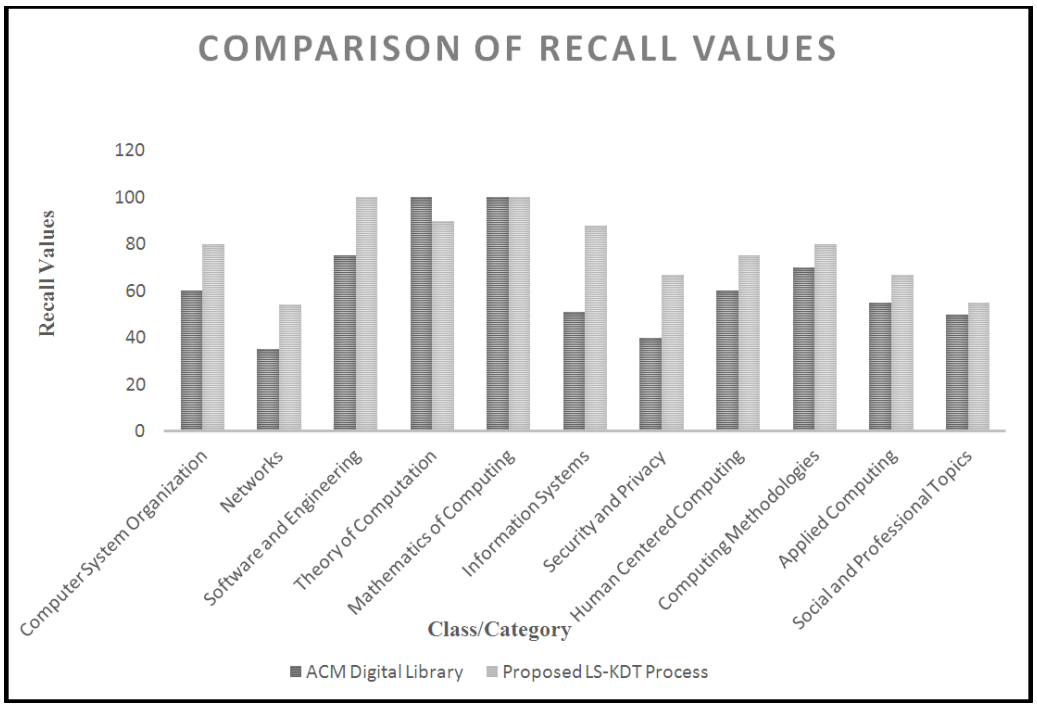


FIGURE 14. Comparison of recall values on computer science articles dataset

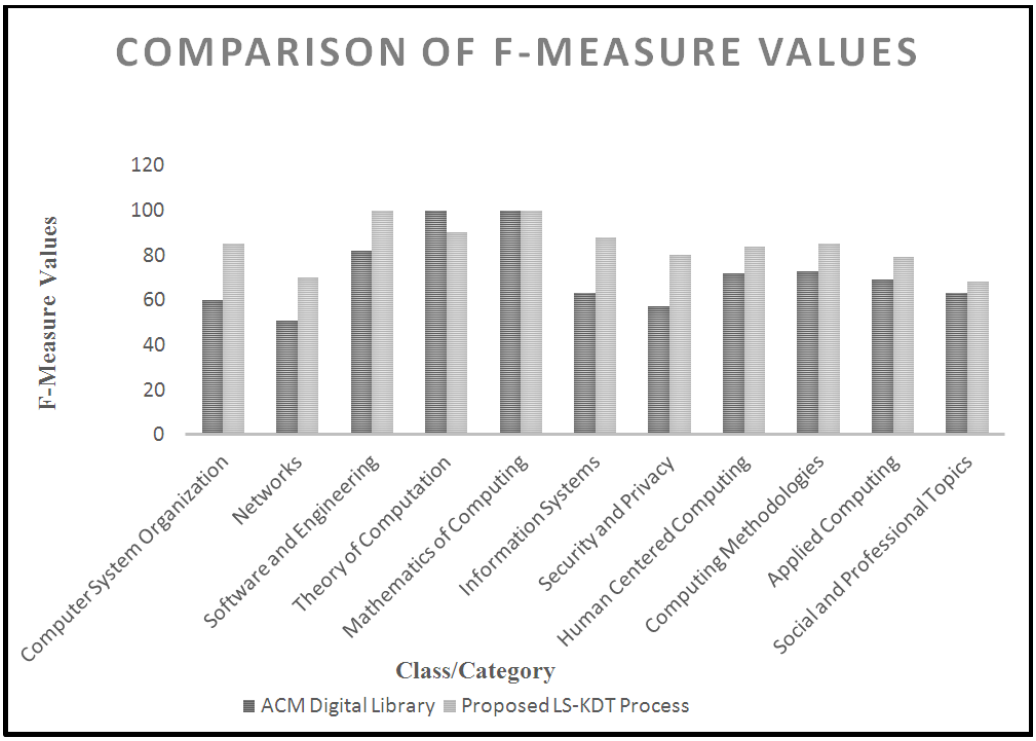


FIGURE 15. Comparison of F-measure values on computer science articles dataset

are improved in our proposed LS-KDT process. The values of precision, recall and F-measure on computer science articles dataset are shown graphically in Figures 13, 14 and 15 respectively.

5.2. **Comparison of results for articles of medical domain.** Table 4 shows the values of precision, recall and F-measure obtained on articles of medical domain. The results are calculated from ACM digital library and the proposed LS-KDT process. The results of precision, recall and F-measure are shown graphically in Figures 16, 17 and 18 respectively.

TABLE 4. Comparison of results for medical domain articles

		ACM Digital Library			Proposed LS-KDT Process		
S. No	**Class/Category**	**Precision**	**Recall**	**F-Measure**	**Precision**	**Recall**	**F-Measure**
1	Computer System Organization	70	60	65	80	80	80
2	Networks	100	45	68	100	54	70
3	Software and Engineering	92	75	83	90	70	79
4	Theory of Computation	100	100	100	90	90	90
5	Mathematics of Computing	95	90	92	100	100	100
6	Information Systems	90	50	64	90	50	64
7	Security and Privacy	100	40	57	100	78	88
8	Human Centered Computing	90	60	72	95	75	84
9	Computing Methodologies	100	90	95	100	90	95
10	Applied Computing	95	65	77	95	67	79
11	Social and Professional Topics	85	50	63	90	55	68
	Average Values	Micro Precision = 92	Micro Recall = 66	Micro F-Measure = 76	Micro Precision = 94	Micro Recall = 76	Micro F-Measure = 82

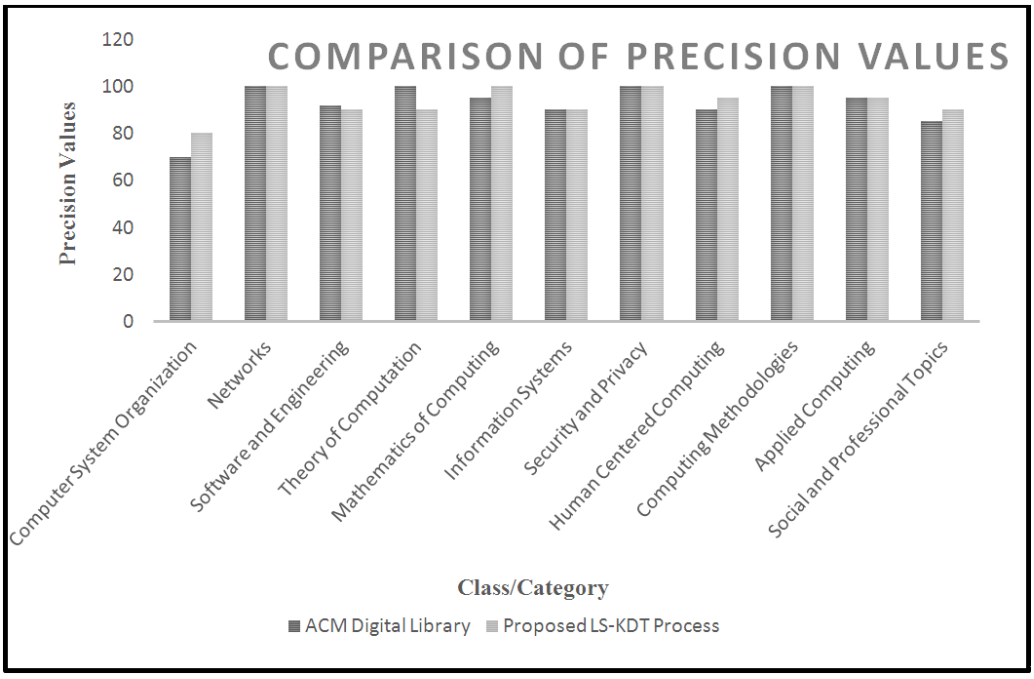


FIGURE 16. Comparison of precision values on medical domain articles

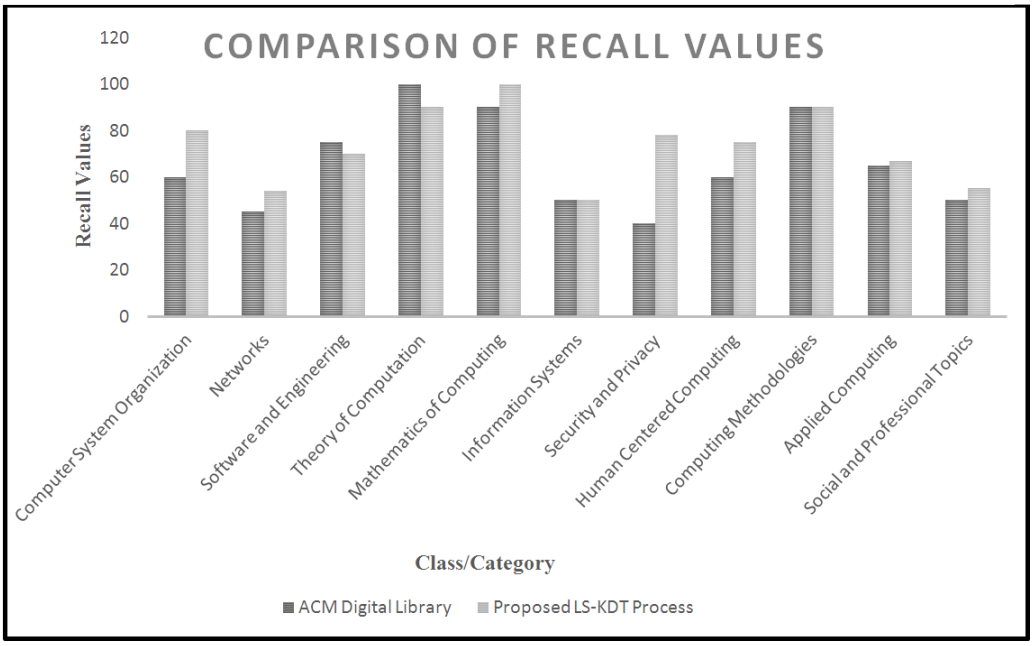


FIGURE 17. Comparison of recall values on medical domain articles

6. **Conclusions.** In this paper, we have proposed a modified LS-KDT process for the automated multi label categorization of text documents. The proposed process has been tested on two datasets of text domain. And the performance is compared with the results of ACM digital library. The standard performance metrics like recall, precision and F-measure are calculated and our proposed process has performed in a significantly better way. In future, this process may be tested on other text datasets like legal documents, and business documents. The results may also be analyzed for other digital libraries and online repositories.

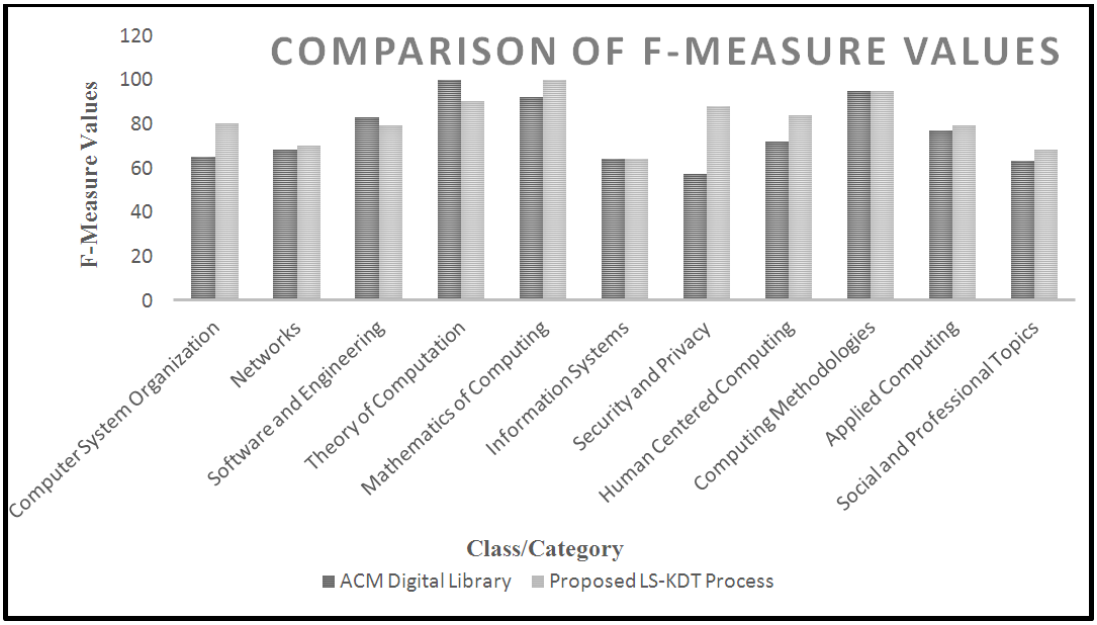


FIGURE 18. Comparison of F-measure values on medical domain articles

REFERENCES

[1] J. Han, M. Kamber and J. Pei, *Data Mining: Concepts and Techniques*, 3rd Edition, Elsevier Publications, 2011.
[2] P. R. Ponniah, *Data Warehouse Fundamentals*, John Wiley & Sons, 2011.
[3] G. Ifrim, *Statistical Learning Techniques for Text Categorization with Sparse Labeled Data*, Ph.D. Thesis, Max-Planck Institute for Informatics, Saarland University, Germany, 2009.
[4] R. Feldman and I. Dagan, Knowledge discovery in textual databases (KDT), *Proc. of KDD-95*, vol.95, pp.112-117, 1995.
[5] R. Jindal and S. Taneja, A lexical-semantics based method for multi label text categorization using Word Net, *International Journal of Data Mining, Modelling and Management*, 2017.
[6] R. Jindal and S. Taneja, Ranking in multi label classification of text documents using quantifiers, *Proc. of IEEE International Conference on Control System, Computing and Engineering (ICCSCE)*, Malaysia, pp.162-166, 2015.
[7] D. G. Rajpathak, An ontology based text mining system for knowledge discovery from the diagnosis data in the automotive domain, *Computers in Industry*, vol.64, no.65, pp.565-580, 2013.
[8] S. Lee et al., Knowledge discovery in inspection reports of marine structures, *Expert Systems with Applications*, vol.41, no.4, pp.1153-1167, 2014.
[9] M. Song et al., PKDE4J: Entity and relation extraction for public knowledge discovery, *Journal of Biomedical Informatics*, vol.57, pp.320-332, 2015.
[10] N. Uramoto et al., A text-mining system for knowledge discovery from biomedical documents, *IBM Systems Journal*, vol.43, no.3, 2004.
[11] R. S. Wagh, Knowledge discovery from legal documents dataset using text mining techniques, *International Journal of Computer Applications*, vol.66, no.23, 2013.
[12] R. Al-Zaidy et al., Mining criminal networks from unstructured text documents, *Digital Investigation*, vol.8, no.3, pp.147-160, 2012.
[13] C. J. Rijsbergen, *Information Retrieval*, 2nd Edition, Butterworth, London, 1979.
[14] G. A. Miller, Word net: A lexical database for English, *Communications of the ACM*, vol.38, no.11, pp.39-41, 1995.
[15] S. Scott and S. Matwin, Text classification using Word Net hypernyms, *Proc. of COLING-ACL'98*, pp.45-52, 1998.
[16] H. Schildt, *Java – The Complete Reference*, 7th Edition, Mcgraw Hill Education, 2007.
[17] H. Lim, J. Lee and D.-W. Kim, Optimization approach for feature selection in multi-label classification, *Pattern Recognition Letters*, vol.89, pp.25-30, 2017.

[18] A. Akbarnejad and M. S. Baghshah, A probabilistic multi-label classifier with missing and noisy labels handling capability, *Pattern Recognition Letters*, vol.89, pp.18-24, 2017.
[19] R. C. Cardoso, F. D. F. D. Souza and A. C. Salgado, Using semantic web concepts to retrieve specific domain information from the web, *Intelligent Information Technologies and Applications*, pp.249-270, 2007.
[20] *www.lextek.com/manuals/onix/stopwords2.html*.
[21] B. G. Mirkin, S. Nascimento and L. M. Pereira, Representing a computer science research organization on the ACM computing classification system, *Proc. of the 16th International Conference on Conceptual Structures (ICCS-2008)*, RWTH Aachen University, pp.57-65, 2008.