CAPITAL UNIVERSITY OF SCIENCE AND TECHNOLOGY, ISLAMABAD

Multi-Label Classification of Computer Science Research Papers Using Papers' Metadata

by

Naseer Ahmed Sajid

A thesis submitted in partial fulfillment for the degree of Doctor of Philosophy

in the

Faculty of Computing

Department of Computer Science

2018

Multi-Label Classification of Computer Science Research Papers Using Papers' Metadata

By
Naseer Ahmed Sajid
(PC-093007)

Foreign Evaluator 1
Dr. Joel Rodrigues, Professor
University of Beira Interior, Bolama, Portugal

Foreign Evaluator 2
Dr. Hermann Maurer, Professor
Graz University of Technology, Austria

Dr. Muhammad Tanvir Afzal
(Thesis Supervisor)

Dr. Nayyer Masood
(Head, Department of Computer Science)

Dr. Muhammad Abdul Qadir
(Dean, Faculty of Computing)

DEPARTMENT OF COMPUTER SCIENCE
CAPITAL UNIVERSITY OF SCIENCE AND TECHNOLOGY
ISLAMABAD
2018

To my family and my supervisor…

CAPITAL UNIVERSITY OF SCIENCE & TECHNOLOGY
ISLAMABAD
Expressway, Kahuta Road, Zone-V, Islamabad
Phone:+92-51-111-555-666 Fax: +92-51-4486705
Email: info@cust.edu.pk Website: https://www.cust.edu.pk

CERTIFICATE OF APPROVAL

This is to certify that the research work presented in the thesis, entitled "**Multi-Label Classification of Computer Science Research Papers Using Papers' Metadata**" was conducted under the supervision of **Dr. Muhammad Tanvir Afzal**. No part of this thesis has been submitted anywhere else for any other degree. This thesis is submitted to the **Department of Computer Science, Capital University of Science and Technology** in partial fulfillment of the requirements for the degree of Doctor in Philosophy in the field of **Computer Science.** The open defence of the thesis was conducted on **13 September, 2018.**

Student Name : Mr. Naseer Ahmed Sajid
(PC093007

The Examination Committee unanimously agrees to award PhD degree in the mentioned field.

Examination Committee :

(a) External Examiner 1: Dr. Ejaz Ahmed
Associate Professor
FAST-NU, Islamabad

(b) External Examiner 2: Dr. Omer Ishaq
Associate Professor
Air University, Islamabad
Omer

(c) Internal Examiner : Dr. Nayyer Masood
Professor
CUST, Islamabad

Supervisor Name : Dr. Muhammad Tanvir Afzal
Associate Professor
CUST, Islamabad

Name of HoD : Dr. Nayyer Masood
Professor
CUST, Islamabad

Name of Dean : Dr. Muhammad Abdul Qadir
Professor
CUST, Islamabad

AUTHOR'S DECLARATION

I, **Mr. Naseer Ahmed Sajid (Registration No. PC093007)**, hereby state that my PhD thesis titled, '**Multi-Label Classification of Computer Science Research Papers Using Papers' Metadata**' is my own work and has not been submitted previously by me for taking any degree from Capital University of Science and Technology, Islamabad or anywhere else in the country/ world.

At any time, if my statement is found to be incorrect even after my graduation, the University has the right to withdraw my PhD Degree.

(Mr. Naseer Ahmed Sajid)

Registration No : PC093007

Dated: 13 September,

PLAGIARISM UNDERTAKING

I solemnly declare that research work presented in the thesis titled "**Multi-Label Classification of Computer Science Research Papers Using Papers' Metadata**" is solely my research work with no significant contribution from any other person. Small contribution/ help wherever taken has been duly acknowledged and that complete thesis has been written by me.

I understand the zero tolerance policy of the HEC and Capital University of Science and Technology towards plagiarism. Therefore, I as an author of the above titled thesis declare that no portion of my thesis has been plagiarized and any material used as reference is properly referred/ cited.

I undertake that if I am found guilty of any formal plagiarism in the above titled thesis even after award of PhD Degree, the University reserves the right to withdraw/ revoke my PhD degree and that HEC and the University have the right to publish my name on the HEC/ University Website on which names of students are placed who submitted plagiarized thesis.

(Mr. Naseer Ahmed Sajid)
Registration No. PC093007

Dated: 13 September, 2018

Acknowledgements

First of all praise be to Allah, The most gracious, Who blessed me with the opportunity, capability and resources to pursue the doctoral program, and it's all due to His grace that I saw it through.

One of the most notable of Allah's blessings upon me was in the form of my supervisor. PhD supervisors need a special skill; they must have night-vision eyes so that when the student gets totally lost in the pitch dark ravines of the research landscape, they can guide the lost soul to the right track. My supervisor has not only got such vision, he's got additional telescopic lens built in. When I announced less than two years and a half ago that I was quitting because I could not see where I was going, he was able to see the path that I needed to traverse in order to reach the summit. I don't have words to thank you, Dr. Muhammad Tanvir Afzal, for the motivation, guidance, support and encouragement that you provided to me on constant basis.

I am grateful to Dr. Muhammad Abdul Qadir, Head of the Center for Distributed and Semantic Computing (CDSC), whose advocacy of the "doer approach" led me to some quick decisions that saved me a lot of precious time. I am also thankful to other members of CDSC whose discussion and constructive criticism maintained an environment that was conducive for research.

There are numerous other people at C.U.S.T. who helped me in pursuit of my PhD in one way or the other including the faculty members at Department of Computer Science, managerial and support staff, and the librarian to mention a few. Thank you all.

I owe a lot to my close friends, Mr. Tariq Ali, Mr. Munir Ahmed, Mr. Sher Afgan and Mr. Tasawer Hussain, for being constant source of inspiration and motivation throughout this time. There are so many other well wishers including friends, colleagues and relations who remembered me in their prayers. Allah blesses you all.

In the end I must mention the being that cares more about me than I do for myself, my mother & father. I always felt their prayers by my side in the time of despair and frustration, and this feeling gave me energy to put myself together. And last but not the least, I have profound gratitude for my wife who had to face double dose of educational stresses: from my children as well as from me. May Allah reward you for your sacrifice and selflessness.

List of Publications

It is certified that following publications have been made out of the research work that has been carried out for this thesis.

Journal Papers

1. Sajid, N.A., Afzal, M.T., and Qadir, M.A. (2016). Multi-label classification of Computer Science documents using fuzzy logic. Journal of the National Science Foundation of Sri Lanka, 44(2), Pp.155–165. [Impact Factor: 0.277]

2. Sajid, N.A., Afzal, M. T., Qadir, M.A., and Khan, S.A. (2013). The Insights of Classification Schemes. Sindh University Research Journal (SURJ 2013), (Science Series), 45 (A-1), Pp. 145-150.

Conference Papers

1. Sajid, N.A., Ali, T., Afzal, M.T., Qadir, M.A., and Ahmed, M. (2011). Exploiting reference section to classify paper's topics. *Proceedings of the International Conference on Management of Emergent Digital EcoSystems (MEDES'2011), San Francisco, California, USA*, Pp. 220 – 225.

2. Sajid, N.A., Qadir, M.A., Afzal, M.T., and Khan, S. A. (2013). Survey of Classifiers based on Different Datasets.*3rd International Conference on Computer and Emerging Technologies (ICCET 2013).*

Naseer Ahmed Sajid
Reg. No PC093007

Abstract

In scientific literature, a publication is deemed to be a way of expression regarding scientific contribution in a specific context of a discipline. It can be further substantiated through a well-known quote that “Communication in science is realized through research publications”. Over the decades, the tremendous increase has been witnessed in the production of documents available in the digital form. The increased production of documents has gained so much momentum that their rate of production jumps two-fold every five years. The large chunk of these documents comprises of research publications due to the subsequent discoveries and inventions in science. This incessant process of research publications has never been interrupted on the contrary, it has gained significant momentum. Almost 28,100 active scholarly journals are publishing almost 2.5 million articles per year. These articles are searched over the Internet via search engines, digital libraries, and citation indexes. However, retrieval of relevant research papers for user queries is still a pipedream. This is due to the fact that scientific documents are not indexed based on some subject classification hierarchies such as ACM classification system for Computer Science. This has motivated researchers to propose innovative approaches for research papers classification. This is not only beneficial for relevant retrieval of research papers but also is helpful in many other application scenarios such as when: (1) journal/conference editors want to identify reviewers; (2) research scholar wishes to identify the suitable supervisor; (3) authors intend to submit their research papers; and (4) one seeks to analyze trends, find experts and to recommend relevant papers etc. In this dissertation, author has critically reviewed the literature on research papers classification and identified the following research deficiencies which have been focused in this dissertation: (1) The existing research papers’ classification schemes utilize content of papers and most of the time, non-availability of content make those schemes non-applicable. There is a need to explore some alternative features to classify research articles that could produce results closer to content based approaches. (2) Majority of state-of-the-art approaches focus on single-label classification, while experiments on comprehensive dataset revealed that a research article may belong to multiple categories. There is a need of such multi-label classification system that utilizes best possible alternate of the content based approaches with closer or improved accuracy. (3) The existing multi-label classification schemes classify citations into limited number of categories, In Computer Science domain; ACM classification system contains 11 classes at its root level. An approach that could classify research articles at least to the root level of ACM classification system is a need of the hour. The objective of this dissertation is to use freely available metadata in the best possible way to perform multi-label classification and to evaluate that; to what extent metadata based features can perform similar to content-based approaches? We have proposed, developed and evaluated techniques on metadata such as Title , Keywords, Title & Keywords, References of the research papers and have reported the achieved results. For classification of research articles based on metadata and into multi-labels, we have harnessed metadata in diverse ways for example: (1) Multi-label Document Classification using Papers' Metadata (Title & Keywords); and (2) Multi-label Document Classification based on Research Articles' References. These techniques have been evaluated for two different and diversified datasets. One dataset is from online journal known as Journal of Universal Computer Science (J.UCS) and other is benchmark dataset comprises of research papers published by the ACM. These techniques yield encouraging results (i.e. 88% of accuracy) by using only freely available metadata as compared to the state-of-the-art techniques on both datasets.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Researchers are completely immersed in discovery of innovative contraptions to minimize human labor. These innovative ideas are being introduced in the form of research publications which is considered a language of scientific communication as further elaborated by Bornmann *et al.* [1], “Communication in science is realized through research publications”. Over the decades, the incredible increase has been seen in the production of documents available in the digital form and is getting doubled every five years [2]. The major part of this plethora of documents comprises on research publications due to the subsequent discoveries and inventions in science [3]. This continuous process of research publications has never been interrupted; on the contrary, it increased rapidly [4]. The most recent report by Ware and Mabe [5], delineates that almost 28,100 active scholarly journals are publishing almost 2.5 million articles per year. These articles are searched over the Internet via search engines, digital libraries and citation indexes. The vast amount of these documents is unstructured in nature, due to which search systems are not efficient enough to retrieve most relevant documents [6]. When user poses a query, the search systems return bulk of documents from which very few documents hold the relevancy to the query. Because of this disorganization of research publications, the problem of classifying research articles into appropriate category has gained the attention of a lot of researchers in document classification community. The researchers are aimed to classify the research articles in such a way that guarantees maximum relevant information retrieval [7]. The availability of this huge corpus on the digital web has made it challenging for researchers to classify these publications into different categories.

How to automatically assign appropriate category to the document or research article? In the late 80's, the document classification was managed by manually building human crafted rules for assigning document to some predefined category. In the 90's, the Machine Learning (ML) paradigm outperformed that manual system, because ML automatically assigns suitable category via supervised learning [7]. To date, numerous approaches have been proposed that perform document classification by using supervised machine learning. These approaches classify documents into different categories [3, 6, 8–10], from which some of the approaches addressed specifically research articles' classification problem [3, 6, 10].

A research article holds an association with particular category or categories. Being specific about the issue of "classification of research articles into predefined set of categories", mapping a research article into the specified category or categories can be beneficial in different scenarios such as: (1) Conference/Journal administrations want to identify reviewers for the submitted papers, (2) Authors want to submit papers in a particular topic of conference, (3) Authors want to search relevant documents to their topics, and (4) Citation indexes and digital libraries want to retrieve relevant papers for user queries. The contemporary research articles' classification schemes broadly rely on two categories: (1) Content based approaches and (2) Metadata based approaches, which are explained in detail in the Chapter 2 of this dissertation.

The content based approaches produced more promising results than metadata based approaches because of their richness in features [11–23]. But what to do when we do not have an access to the content? Major journal publishers like: ACM, Springer, Elsevier, IEEE etc. do not provide open access to their articles as there are financial, legal and technical barriers. In such scenarios: there should be an alternative way to classify research articles. Such best alternate is available in the form of freely available metadata. Metadata is defined as data about data or some external information about the actual data. Various kinds of useful research articles' metadata such as, title, authors, keywords, categories etc. are almost freely available online.

This dissertation focuses on classification of research articles from Computer Science domain via exploitation of freely available metadata such as, title, author, keywords and categories. To address this issue, we have also picked the reference section of articles due to the following reasons: (1) Consider a scenario where you have an article that belongs to specific topic, and you want to acquire more research articles of the same topic. The best possible way would be the citations (A citation in a research paper is a reference to a published or unpublished resource that you consulted and obtained information from while writing your research paper) exploitation of that particular topic because citation delineates a relationship between a part and the whole of the citing document [24]. (2) Most of the time, cited and citing work lie under the same category.

1.1 What is Classification

In Machine Learning (ML), classification is regarded as a central concept that aims to classify items into two or more groups. The classification is performed on various ML problems, for instance: speech recognition [25], text categorization [26, 27], etc. In scientific literature, document classification is beneficial to retrieve useful information [28]. Usual method of document classification comprises on the selection of useful features from data which could help to assign some target category. The classification can be of two forms: (1) single-label classification (i.e., classifying the items into single class) and (2) multi-label classification (i.e., classifying the items into more than one class). Since, a research article can have an association with multiple categories as explained with the help of examples in Chapter 3 (Table 3.2). Therefore, multi-label classification has gained the attention of many researchers where they have classified research articles into multiple categories [21–23]. Most of the multi-label classification schemes produce low accuracy and classify research articles into limited no of categories [14, 29, 30]. The classification of research articles into multiple categories with high accuracy is a challenging task [31]. Of course, multi-label classification requires immense effort to produce diversified and comprehensive set of features that specifically belong to each category. This research work specifically focuses on multi-label classification of research articles by using only metadata of the research papers and achieved accuracy close to the state-of-the-arts content based approaches.

1.2 ACM Classification System

The Association for Computing Machinery (ACM) categorization system is commonly used for organizing research papers belonging to the Computer Science domain into topical taxonomy defined by the ACM. In 1964, first ACM classification system [32, 33] is introduced in the domain of Computer Science for the

Table 1.1: ACM Categories (CCS98)

Levels	**Categories**
1	11
2	81
3	400

classification of scientific documents. The ACM published an entirely new system in 1982. Based on 1982 system new versions are published in years 1983, 1987, 1991 and 1998. The ACM 1998 classification system considers a de facto standard classification system in the Computer Science. In 2012, new ACM system is developed and the old schemes are mapped into this new system. Both the 1998 and 2012 systems are available on Citation Pages of all indexed articles in the ACM Digital Library[1] . There are three levels of ACM Computing Classification System 1998 (CCS98) [18, 32, 34].

In Figure 1.1, Level 1 represents topics from A (General Literature) to K (Computing Milieux) and it contains total 11 topics (Table 1.1). At second level, each topic of Level 1 has sub-topics, for example, for topic "C (Computer System Organization)", there exists C.0 (General), C.1 (Processor Architectures), C.2 (Computer Communication Network),. . . ., C.m (Miscellaneous) topics and total second level topics are 81. Similarly at the third level, each topic of Level 2 has sub-topics such as C.2.0 (General),C.2.1 (Network Architecture and Design), C.2.2 (Network Protocols),.., C.2.m (Miscellaneous), and total third level topics are 400.

This dissertation focuses on mapping research articles belonging to the Computer Science domain into ACM category or categories at root level only (i.e. 11 topics). We have utilized 1998 version of ACM categorization system due to the following two reasons. (1) ACM 1998 classification system is being used by ACM digital library for annotating research papers into this list of topics; (2) We have compared our results with the proposed technique by Santos and Rodrigues who have utilized 1998 ACM version [18]. This research work scrutinizes to classify research articles into maximum appropriate categories of this ACM categorization system. Figure 1.1, visualizes the overview of ACM categorization system.

[1] http://www.acm.org/about/class

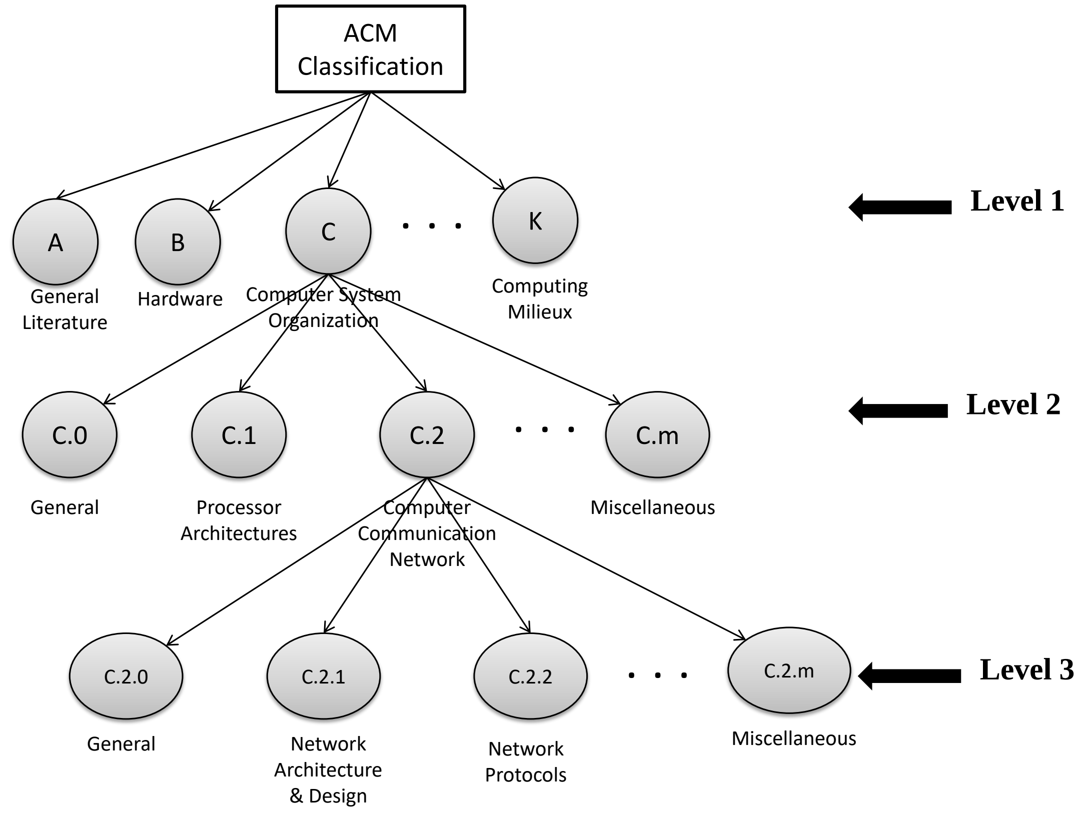


Figure 1.1: ACM Classification Hierarchy

1.3 Overview of State-of-the-Art Approaches

This section encompasses brief overview of state-of-the-art approaches which provides a fair idea about the current trends in research articles' classification community. The detailed explanation of state-of-the art approaches is reported in Chapter 2. The contemporary approaches that address the issue of research articles classification are broadly categorized into two chunks: (1) Content Based Approaches and (2) Metadata Based Approaches. The Content Based Approaches perform classification by harnessing different features relying on content of research articles [11–23]. These features can be in the form of important terms or phrases (often referred as cue words or cue phrases) from research articles. For research articles classification, different measures are applied on the content of research papers for example, normally the important terms are extracted using Term Frequency-Inverse Document Frequency (TF-IDF) and Cosine Similarity is computed between the weights of the extracted terms from TF-IDF. The obtained score from these measures are then assigned for supervised learning to predict class of each research article. In literature, very few approaches utilize metadata of research articles for classification. The existing state-of-the-art metadata based approaches exploit the external information about the research articles such as title, authors, keywords, categories etc. and also some content based features to classify research articles into different categories but with low accuracy [13, 14, 35]. We have observed different points from literature that have led us to formulate our research problems. Let's discuss them step by step:

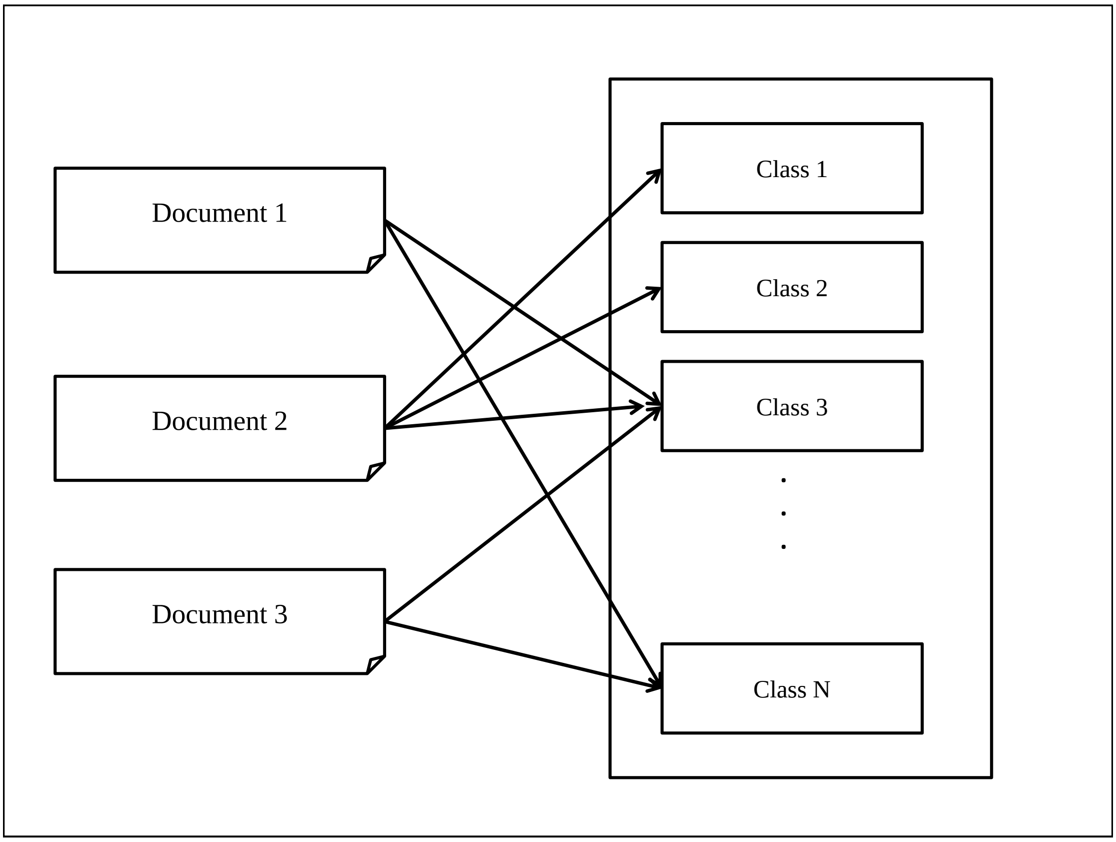


FIGURE 1.2: Multi-Label Classification

- Most of the state-of-the-art approaches performed single-label classification of research articles [12, 16, 17, 19–21, 41–44, 47, 50, 61]. However, research articles may belong to multiple classes as shown in Figure 1.2. For example a paper working on "Network Routing Algorithm" belongs to two classes such as: "Network" and "Algorithm". The comprehensive experiments on such analysis have been provided in Chapter 3 (Table 3.2).
- Most of state-of-the-art approaches utilized content based features for single class or multi-label classification of documents [13, 14, 18, 34, 35, 54].
- In case of multi-label classification, the existing schemes classified articles into small number of categories.

- State-of-the-art approaches had utilized a few numbers of classes.

- Currently no scheme exists that performs multi-label classification by relying fully on freely available metadata.

1.4 Problem Statement

We have formulated our problem statement on the basis of above mentioned observations which are as follows:

1. The existing research articles' classification schemes depend upon content of the articles. Most of the time, non-availability of research articles makes those schemes non applicable. There is a need of some best alternative ways to classify research articles that produce results closer to or better than content based approaches.

2. Majority of state-of-the-art approaches focus on single-label classification, while experiments on comprehensive datasets performed in Chapter 3 (Table 3.2) revealed that a research paper may belong to multiple categories. There is a need of such multi-label classification system that utilizes best possible alternate of the content based approaches with closer or improved accuracy.

3. The existing multi-label classification schemes classify research articles into limited number of categories. While a research article may belong to multiple categories, for instance, in Computer Science domain, the research articles may belong to more than one category of ACM classification system. The ACM categorization system has 11 topics on its root level. The contemporary approaches have only experimented with a few number of classes; however, in real scenarios a paper needs to be classified in any of the 11 ACM topics. Therefore, the focus of this dissertation is to classify research articles to the root level.

1.5 Objectives

The objectives of this dissertation are as follow:

1. The first objective is to classify research articles by using freely available metadata instead of using the whole content of the articles.
2. The second objective is to perform multi-label ACM classification (only root level topics) instead of only performing single-label classification by using metadata of the research articles.
3. The third objective is to evaluate that to what extent metadata based features can perform close to content-based approaches? Moreover, how much the scheme is useful for multi-label classification?

1.6 Research Contributions

We have critically reviewed various state-of-the-art approaches of document classification and proposed two novel multi-label classification approaches that rely completely on freely available metadata. Let's discuss these approaches step by step:

1. The first novel approach exploits metadata of research articles for multi-label classification. The metadata includes Title and Keywords of research paper. The experimental results of this approach have been published in the journal named: "Journal of National Science Foundation Sri Lanka" in 2016 [36]. For evaluation of this approach, we have used two datasets such as: J.UCS dataset [37] and the comprehensive dataset constructed by Santos *et al.* [18]. The J.UCS dataset contains 1460 research papers. There are two reasons for selecting J.UCS dataset (1) the J.UCS covers all areas of Computer Science topics; (2) the authors belong to diversified domains, which give a fair chance to the proposed technique to evaluate the system. Similarly, the reason for the selection of ACM dataset is that it contains research publications from different conferences, journals and the workshops and it contains 86,116 research papers. Furthermore, this dataset has been constructed by the state-of-the-art approach [18] which will enable us to

compare with the best known approach. We have extracted Title, keywords and Categories from these research articles. These metadata parameters are picked because they demonstrate the theme of research work. The research papers are mapped into their appropriate category or categories on the basis of term frequency weights that follows the defined threshold criteria. The optimum threshold value has been founded by performing experiments on different and diversified datasets. Surprisingly, at optimum threshold value of 0.2 and for metadata (Title), our results outperformed on the same dataset used by Santos *et al.* [18] by obtaining accuracy of 0.85, precision of 0.95, recall of 0.88, and F-measure of 0.89. The detailed explanation of this approach is described in Chapter 3.

2. The second multi-label novel research articles' classification approach exploits references of research articles. This approach has been published in ACM conference in 2011 [10]. The reference parameter is picked based on the assumption that citing work cites articles from the same/similar topics. From the datasets, we have generated Topic-References (TR) pairs against pre-defined ACM categories. The system collects corresponding list of topics matched with the references in the said pair. Subsequently, multiple weights are assigned during the process of this matching. The appropriate category or categories of research paper is suggested based on defined threshold values which has been founded after experiments on diversified datasets. Our approach is able to predict top level categories of ACM classification system with 74 % accuracy. The detailed explanation of this approach is described in Chapter 4.

3. In this multi-label classification, we are also interested to know the performance of different state-of-the-art classifiers. In 2013, we have published an article comprises on classifiers performances in "Science Series Journal" [38]. In this article, we have harnessed different classifiers on variety of datasets such as Vote, Weather, Super Market, Diabetes, Contact Lenses, Iris and Labor by using Waikato Environment for Knowledge Analysis (WEKA[2]) tool. These experiments provided us enough understanding about the need of specific classifier for specific type of data. This research gives brief insights of classification schemes commonly used in Machine Learning (ML). Each technique has its own merits and demerits and can be incorporated

[2] http://www.cs.waikato.ac.nz/ml/weka/

depending on the situation. Some of these may be valuable but others may not in the same situation and vice versa. When new instance is available then these techniques may help to classify new instance into different classes. According to the nature of dataset, classifier may be selected for the classification of the instances. We have incorporated this understanding in this dissertation via using appropriate classifiers and comparing their results.

1.7 Research Methodology

From the application point of view, the research presented in this dissertation can be termed as applied research: Instance solution to the problem is proposed and evaluated. The document classification approaches used in this research are mostly quantitative in nature; experiments have been performed and results are evaluated to form the idea about the results. For conducting this research, Kumar [39] model has been followed with the slight changes as per the requirements of this research. The activities carried out during this research are described below and shown in Figure 1.3.

1.7.1 Phase I - Deciding Scope of Research

Step 1: Formulating a research problem: This step contains three tasks;

1. Literature review.
2. Definition of criteria for evaluation of document classification techniques.
3. Observations from the literature review.

Step 2: Proposal of new Metadata-based Multi-Label Document Classification Techniques: Proposal for devising and evaluating novel multi-label document classification techniques was formulated as the next step after identifying the research problems in the literature.

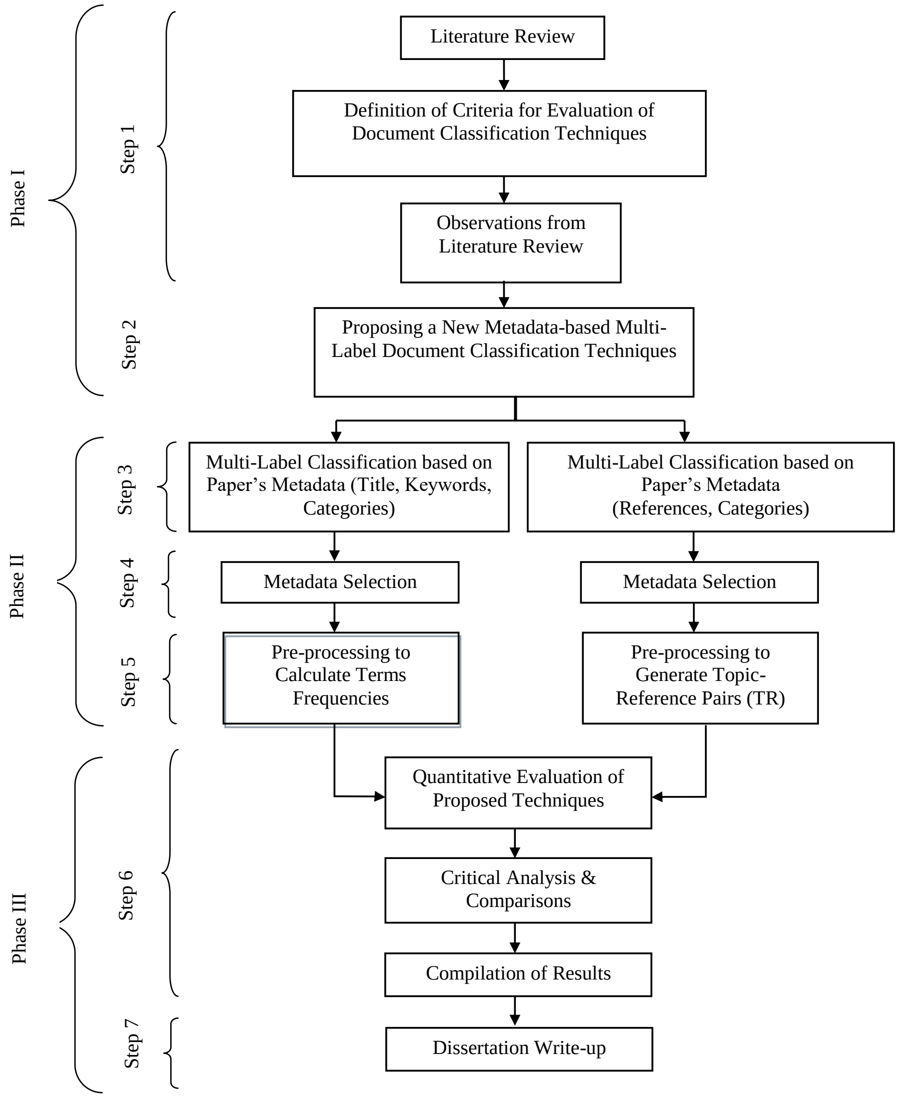

Phase I
Step 1
Step 2
Literature Review
Definition of Criteria for Evaluation of Document Classification Techniques
Observations from Literature Review
Proposing a New Metadata-based Multi-Label Document Classification Techniques
Phase II
Step 3
Step 4
Step 5
Multi-Label Classification based on Paper's Metadata (Title, Keywords, Categories)
Multi-Label Classification based on Paper's Metadata (References, Categories)
Metadata Selection
Metadata Selection
Pre-processing to Calculate Terms Frequencies
Pre-processing to Generate Topic-Reference Pairs (TR)
Phase III
Step 6
Step 7
Quantitative Evaluation of Proposed Techniques
Critical Analysis & Comparisons
Compilation of Results
Dissertation Write-up

FIGURE 1.3: Research Methodology

1.7.2 Phase II – Planning the Research Study

Step 3: Conceptualizing a Research Design: This research was divided into two proposed approaches which will be discussed in the Chapter 3 and Chapter 4.

Step 4: Constructing an Instrument for Metadata Data Selection: Instead of creating a new dataset, a freely available large and comprehensive datasets were acquired which has also been used in previous research.
Step 5: Selecting required Metadata after Preprocessing: Computing the terms frequency weights for metadata based approach and generating topic-references pairs for the references based approach.

1.7.3 PHASE III - Conducting the Research Study

Step 6: Quantitative Evaluation: Evaluation of proposed techniques, critically analyzing the results, comparison of the results with the state-of-the-art approaches and its discussion.
Step 7: Dissertation write-up: This dissertation is the output of all activities performed in all the previous steps as described.

1.8 Organizing the Dissertation

This dissertation is organized as follows. Chapter 2 sheds light on research efforts carried out in the domain of document classification and critical review of the state-of-the-art approaches proposed in the literature. Chapter 3 focuses on multi-label document classification approach based on the research papers' metadata (*Title* and *Keywords*) with its results and discussion about the results. Chapter 4 highlights another multi-label document classification approach based on the *Reference's Section* of the research papers with its results and discussion. The critical analysis and comparisons of proposed approaches' results with the state-of-the-art approaches and discussion about the results are presented in Chapter 5. Chapter 6 summarizes and concludes the dissertation and also highlights the future directions.

Chapter 2

Literature Review

The preceding chapter (Chapter 1) sheds enough light on the scenarios that have led us to formulate the problem statement along with its scope. This chapter focuses on the detailed overview of state-of-the-art approaches as every scientific study is dependent upon the study of erudite peers in the field. The document classification community is focused on proposing innovative ideas for document classification as the number of documents in digital form is increasing. Text classification is a very old dilemma. As early as the 1800s, studies were done on verifying the authorship of the works of Shakespeare [40]. When the first document classification approach was proposed thereafter the process started to emerge into different branches as a result whereof community began classification of specific type of documents for instance: (1) magazine (2) newspapers etc [16, 21, 41–43]. Since then document classification community has diverted its attention specifically on the research papers classification due to subsequent inventions in scientific literature. The contemporary approaches that address the issue of research articles' classification broadly rely on two categories. (1) Content based approaches (2) Metadata based approaches. Let's discuss these categories step by step.

2.1 Content Based Approaches

Currently, the document classification community is kind of biased when it comes to data exploitation of research papers in order to categorize or classify them. Most of the contemporary approaches rely on the content of research articles due to richness in features which can be constructed by exploiting the whole content.

This section focuses on content based state-of-the-art approaches.

In 2016, Tang *et al.* [44], proposed a novel Bayesian automatic text classification approach by exploiting different content based features. They proposed a class dependent set of features. They formulated classification rules by harnessing Baggenstoss's PDF Projection Theorem for the conversion of class-specific PDFs in low-dimensional feature into raw data space. They have evaluated their approach on two different real-world benchmark datasets. 1) The first dataset is of 20-NEWSGROUP that comprises of 20,000 online posted documents belonging to 20 different topics and 2) The second dataset "REUTERS comprises of 21, 578 documents belonging to 135 topics. Their system achieved remarkable accuracy for 20-NEWSGROUP dataset; however, they did not mention the exact figures. They have also presented another approach based on feature selection framework for the Naïve Bayes [41]. These selected features are ranked for the classification. They presented a new divergence measure which is called "Jeffreys-Multi-Hypothesis (JMH) divergence, to measure multi-distribution divergence for multi-label classification". Two features selection methods were developed by using the JMH divergence and achieved accuracy of 0.95 and F-Measure of 0.90.

Another study by Shedbale *et al.* [42], is based on the survey of features selection approaches for text classification. They have highlighted the existing feature selection schemes and different methods of reducing the dimension of these features. These methods are categorized into two categories 1) wrapper and 2) filter. The filter scheme provides significant performance over wrapper scheme without classifier's feedback. The filter scheme has been used in most of the text classification problems.

In 2016, Zhou *et al.* [45] built a content based classifier by using Naïve Bayes and Logistic Regression algorithms. They have used annotated datasets of CiteSeerX and arXiv belonging to the Computer Science domain. The classifier is built by relying on different features from which Bi-grams feature outperformed for both datasets. The F1-Measure for arXiv dataset and CiteSeerX dataset are 0.95 and 0.76 respectively. Similarly, Zong *et al.* [43] approach classifies research papers on the basis of different features by applying semantic similarity on them. The experimental results are evaluated on two datasets Routers-10 and 20-Newsgroups. By applying the SVM algorithm they achieved the F-Measure of 0.76 for Newsgroup dataset and 0.91 for Routers dataset.

Another content based classification and visualization of scientific documents is proposed by Giannakopoulos *et al.* [23]. This approach has been trained on three different datasets and their categorizations. All modules of this approach have been implemented by using the madIS system. The madIS system provides data evaluation functionalities via extended relational database. The automatic clustering approach of scientific text and newspapers articles was proposed by Afonso and Duque [46]. These articles are taken from Brazilian Portuiguese. A content level approach was proposed by Dendek *et al.* [22] for the classification of scientific documents. They applied different algorithm like: Naive Bayes, decision tree, k-Nearest neighbor (KNN), neural network, Support Vector Machine (SVM) for the classification of documents.

Yaguinuma *et al.* [47] proposed a fuzzy ontology to represent and reasoning over fuzzy or vague information. Arash and Mahdi [20] presented an automatic subject indexing approach for digital libraries and repositories. They proposed a concept matching approach by identifying these concepts from the documents. Then concept similarities are computed with the documents. Concept similarity is used for the classification of documents. Hingmire *et al.* [21] proposed document classification algorithm based on the Latent Dirichlet Allocation (LDA) [48] and unlabeled dataset. The algorithm of this approach assigns one topic to one class label. A query extension method is proposed by Ortuño et al. [49], which extends information related to research papers by using its cited references. Evaluation of this approach was conducted on biomedical documents of the PubMed database.

Another content based hierarchical classification technique of textual data is presented by Duwairi and Al-Zubaidi [50]. They proposed a classifier that is based on a modified version of the well-known K-Nearest Neighbors classifier (K-NN). The original classifier works only with the category representatives instead of the training documents. This category representation saved their effort and time as they did not need to deal with all training documents and categories of different levels. They concluded that there is a need of an effective feature selection technique with the diversified dataset for the text classification [51, 52]. Galke et al. [53] presented a systematic evaluation of classification approaches to explore how far semantic annotations can be conducted using just metadata of the documents. The evaluation has been done with the classification obtained from analyzing only the metadata and with analyzing the semantic annotation of the whole text.

Santos and Rodrigues [18] proposed an approach to assign a scientific document to one or more classes which is called multi-label hierarchy by using the content of the scientific documents. They have extracted these scientific documents from the

ACM library which contain scientific documents from different workshops, conferences and journals from the domain of Computer Science. After the performance analysis of different classifiers, they concluded that combination of both, the collection size (5,000 and 10,000 documents) and numbers of different terms used have no importance for multi-label hierarchical document collection. Similar approach had been presented by Lijuan [54], on the basis of ranking category relevance to evaluate the multi-label problems. Author also proposed an automatic learning approach for the classification of the documents. Li *et al.* also proposed an automated hierarchy approach for document classification [16]. They have utilized the linear discriminate projection method to generate intermediate level of the hierarchy. In this approach, all documents are first transformed into low dimension space and then classified according to the proposed hierarchy. Another similar approach is proposed by Wang and Desai [34] for the CINDI[1] digital library. They formulated their method for ranking classes on the same level which can be helpful for text classification. The evaluation of this approach has been done on the collected dataset by using ACM98[2] classification scheme. Cai and Hofmann [55], presented another hierarchical approach to classify text document by using SVM classifier. They exploited the relationships among the classes which are commonly expressed in the form of hierarchy. Yan et al. [56] proposed a multi-label documents ranking model based on Long Short Term Memory(LSTM). It consisted of two processes one was repLSTM (an adapted representation process) and other one was rankLSTM (a unified learning ranking process). Three datasets have been used for the experiments to classify documents with reasonable performance of their proposed model. Wang et al. [57] proposed an ensemble classification method which groups together random forest and semantic core co-occurrence latent semantic vector space (CLSVSM). Yahoo dataset has been used for the experiments which revealed effectiveness of the proposed method with reasonable results. Baker and Korhonen [58] presented a method which performed hierarchical multi-label document classification by initializing a neural network model. They evaluated their approach on biomedical domain using both sentences and document level classification.

[1] https://cindi.encs.concordia.ca

[2] http://www.acm.org/about/class/ccs98-html

Senthamarai and Ramaraj [17] proposed a technique for classification of text documents based on the text similarity. However, they have implemented their approach on the small amount of dataset. They presented a feature selection framework which calculates the score of selected words for text classification. They have also presented a learning model for text categorization, in which document collections were randomly selected and annotated by the domain experts. After the performance analysis they concluded that smaller vocabulary can accelerate the classification process. They did experiment on the Reuters dataset which contains 21,578 documents and compared the results against different classifiers such as SVM, Rocchio algorithm, Bayes, Naive Bayes etc. However, they have not reported the outcomes of these comparisons. Another hierarchical content based technique is proposed by Wang and Desai [34] in which they have extracted research articles from ACM digital library belonging to the Computer Science domain. They have narrowed down their problem with properties such as all leaf nodes have real categories, multi-label classification and a tree like classification scheme. Their method of text classification specifies and prepares rank for the categories at the same level. Their method works from top to down in hierarchy until the suggested category is assigned. They have used flat local multi-label classifier which serves basic block in their hierarchical classification system. To achieve effectiveness results, they choose Naïve Bayes and Centroid Classifier. They also described the re-ranking process to assign a category at the same level except at the level 0. The evaluation of this classification approach is also presented by Brucher [59]; they evaluated different classification approaches with their merits and demerits. Another approach for automatic documents classification has been presented by Goller *et al.* [11]. They evaluated different approaches on document classification for German text. Their results highlighted the significance of features selection in order to avoid the over fitting problem.

The document classification community is dominated with the content based approaches. Off- course, these approaches have richness in features and produce promising results. To make these schemes applicable the content of the documents is a vital requirement but most of the digital libraries are subscription based like ACM, IEEE, and Springer etc. There is a need of some alternate methods to categorize documents when the content is not available. Such alternate is available in the form of metadata like authors, title, keywords etc. To date, there are very few document classification approaches that exploit metadata of research articles. Let's discuss metadata based document classification approaches in next section.

2.2 Metadata Based Approaches

The contemporary metadata based state-of-the-art research articles classification schemes exploit the metadata of research articles for their classification into a pre-defined hierarchy. Metadata of scientific documents includes title, authors, keywords, categories etc. Such form of metadata is almost freely available online as compared to the whole content of the articles. This section focuses on the brief overview of the metadata based approaches.

A natural language processing approach was proposed by Yohan *et al.* [60] in which they identified named entities and classified them into their different categories. They proposed a rule based system for classification and recognition of named entities in Teluge language. They have utilized features based on word, context and lookup level for classification and detection of named entities. This system is evaluated on different corpus of Teluguwiki and newspaper and was also evaluated for full sentences data to identify named entities with precision range from 0.79 to 0.94.

Another metadata extraction scheme is proposed by Flynn [35], for the document classification. Author proposed “post hoc” classification system for the document classification. After the metadata extraction of the document, the post hoc technique applied further to classify these documents. The experiments are performed on the defense technical information center (DTIC) dataset which contains more than one million documents of various forms like scientific articles, PhD synopsis, conferences papers, slides, public law documents etc. This technique classifies documents correctly by 0.83 of the time.

Khor and Ting [14], proposed a framework by using the Bayesian Network (BN). For classification, four hundred (400) conference papers were collected and classified into four major topics such as: Intelligent Tutoring System, Cognition, e-Learning, and Teacher Education. They have used the keywords of research papers for classification. They have implemented 80-20 split of collected papers, 80% of the papers are used for keywords extraction and BN parameter learning whereas the other 20% are used to predict accuracy performance. A feature selection algorithm is applied to automatically extracted keywords for each topic. The construction of BN is done by using these extracted keywords. For this purpose they proposed a keywords selection algorithm. Pre-processing has been done on the extracted keywords to normalize each keyword by removing stop words and applied

Porter Stemmer [9] algorithm. The Bayesian Network (BN) is a probabilistic reasoning graphical model. For structural representation of variable in the domain, the Directed Acyclic Graph (DAG) is used and can be described by using direct probabilistic dependencies among the variables. The Bayesian Network is used to perform prediction and diagnose reasoning. By comparing the predictive accuracy of human experts classification, BN learning and Naïve Bayesian, efficiency of the BN can be calculated. From the experiments they concluded that BN had an average accuracy of 0.84. The network has been used through a series of validation by human experts and experimental evaluation to analyze its predictive accuracy. The proposed BN has outperformed Naïve Bayesian Classifier. However, all the research articles do not have the keywords as all authors do not provide keywords in their articles. From our point of view, this technique only considered those documents which contained keywords' section. To improve the performance of document classification approaches into predefined categories, Zhang *et al.* proposed another approach [13]. In this approach, they combined citation information and structural contents like title, abstract of the documents. Different similarity measures based on the structural contents and citation information are evaluated to improve the effectiveness of the classification. For this purpose, they extracted documents from ACM Digital Library and used Genetic Programming (GP) approaches to classify these documents into pre-defined ACM subject hierarchy.

To address the document classification problem, the researchers have employed different schemes on two data sources such as: metadata and content. The content based schemes exploit the content of research articles for their classification [11–14, 16–23]. Every scheme has its own pros and cons which depends on the size, pre-processing and nature of the dataset. For these schemes implementation, the content of research articles is an essential requirement. The content based schemes provide better precision due to rich number of features [22]; however, content of the scientific documents is not freely available most of the time. On the other hand, very few researchers have used only the metadata of the documents for the classification [13, 14, 35]. Metadata of the documents provides limited number of features which may result into low accuracy as compared to the content based document classification schemes.

The objective of this dissertation is to use freely available metadata and to analyze that to which extent the metadata based features can behave like content based features. Moreover, how much the scheme is useful for multi-label classification?

The metadata is freely available in majority of scientific digital libraries like IEEE[3] , ACM[4] and Springer[5].

2.3 Evaluation Criteria

For comprehensively understanding the critical findings of the literature, this section has defined evaluation criteria on which all key papers from the literature has been evaluated and has been shown as a comparative study in the Table 2.1– 2.3.

2.3.1 Type of the Data

First evaluation criterion is the type of data, researcher from the diversified domain have exploited data sources like metadata and content of the documents. Some researcher used metadata of the documents and most of the researchers used the content of the documents.

2.3.2 Classification Type

A second criterion is the classification type. The single class means that we have many classes but one document will be classified into only one class. The multi-label (multi-class) means we have many classes and one document may be classified into one or more than one classes. The experiments on diversified dataset (see Table 3.2) show that there are various research papers that may belong to multiple classes.

2.3.3 Number of Classes

Next evaluation criterion is the number of different classes. This highlights that a particular research paper belongs to how many classes. As we have described earlier (see Chapter 1, Section 1.2) that the standard classification scheme is ACM, which contains 11 topics on its root. However, the researchers have not used

[3] http://ieeexplore.ieee.org/

[4] http://dl.acm.org/citation.cfm?id= 2077531

[5] http://link.springer.com/chapter/10.1007%2F11925231_98

all these 11 topics to evaluate their approaches; they have used limited number of ACM topics for categorization.

2.3.4 Dataset

The evaluation criterion of dataset will depict that how many documents are used to evaluate the approaches from the literature. This will highlight the average number of documents we should pick for our experiments for the evaluation of the proposed approaches.

2.3.5 Algorithm / Methodology

This criterion will discuss the algorithms and methodologies used in the literature for the evaluation of the research documents. This will further help us to form an evaluation and comparison strategy.

2.3.6 Evaluation Parameters

This evaluation criterion will highlight that which scientific documents have used which evaluation parameters for example: Accuracy, Precision, Recall and F-Measure.

2.3.7 Results

The last criterion is the results, which will demonstrate that how much value of accuracy/precision/recall has been achieved so far in contemporary state-of-the-art approaches.

2.4 Critical Analysis of Contemporary Approaches based on Evaluation Criteria

After the comprehensive analysis of the above mentioned state-of-the-art approaches, we have concluded that these classification approaches exploited different data sources like some of these have used metadata while others have used content of the documents. Based on the above discussed observations, we have classified state-of-the-art approaches into three types: 1) Content based approaches which exploit content data source and classified documents into only one class (Single-label) from the multiple classes. 2) Content based approaches which exploit content data source and classified documents into one or more than one class (Multi-label). 3) Metadata based approaches which exploited metadata data source of the documents and classified documents into either single-label or multi-label from the multiple classes.

2.4.1 Analysis of Content Based Approaches (Single-label Classification)

The state-of-the-art approaches which exploited content of the documents are shown in the Table 2.1. Researchers of these approaches performed content based document classification and classified documents into only one class. Different algorithms were used to predict the most relevant class. Similarly, different datasets were used for the classification of documents. For example, datasets 20-Newspapers and Routers were used by many researchers [16, 19, 21, 41–44]. Similarly, some other datasets have also been used for the document classification [12, 17, 20, 47, 50, 61]. From Table 2.1, we can examine that the number of classes vary from dataset to dataset.

Different researchers classified documents into different number of pre-defined classes. By using these datasets and pre-defined number of classes, variety of state-of-the-art approaches were presented in the last couple of decades and exploited content of the documents to classify documents into single-label. These approaches have used different evaluation parameters like accuracy, precision, recall and F-measure. These approaches achieved accuracy from 0.4 to 0.95 by exploiting the content of the documents. Similarly, parameter precision achieved from 0.61 to 0.8, Recall achieved from 0.55 to 0.76 and parameter F-measure achieved from 0.71 to 0.94 as mentioned in the literature. These values are significantly good because these techniques have exploited the content of documents which contain huge bag of words (features) for the classification.

Table 2.1: Critical Analysis of Content based Approaches for Single-Label

S.No.	Approaches	Type of Data	Classification Type	No. of Classes	Dataset	Algorithm / Methodolgy	Evaluation Parameters	Results
1	Galke et al., 2017	Content	Single-Label	Econ (4), Polite (5), RCV1(14), NVT (2)	Econ (62,924), Polite (27,576), RCV1(100,000), NVT (100,000)	KNN	F-Measure	Econ (0.41), Polite (0.27), RCV1(0.76), NVT (0.40)
2	Tang et al., 2016a	Content	Single-Label	20-Newsgroup (20), Reuters (135)	20-Newsgroup (20,000), Reuters (21,578)	Naive Bayes	Accuracy, F-Measure	Accuracy (0.095), F-Measure (0.90)
3	Tang et al., 2016b	Content	Single-Label	20-Newsgroup (20), Reuters (135)	20-Newsgroup (20,000), Reuters (21,578)	Bayesian	F-Measure, G-Mean	Not Reported
4	Shedbale et al., 2016	Content	Single-Label	C, Reuters (135)	20-Newsgroup (20,000), Reuters (21,578)	Survey	Accuracy, F-Measure	Accuracy (0.095), F-Measure (0.90)
5	Zhou, 2016	Content	Single-Label	Not Reported	CiteSeerX (665,483), arXiv (84,172)	Naive Bayes, Logistic Regression	F-Measure	CiteSeerX (0.76), arXiv (0.95)
6	Zong et al., 2015	Content	Single-Label	20-Newsgroup (20), Reuters-10 (10)	20-Newsgroup (16,391), Reuters-10 (7,224)	SVM	F-Measure	20-Newsgroup (0.76), Reuters-10 (0.91)
7	Yaguinuma et al., 2014	Content	Single-Label	4	100 Documents	Fuzz-Onto	Accuracy	Accuracy (0.44)

S.No.	Approaches	Type of Data	Classification Type	No. of Classes	Dataset	Algorithm / Methodolgy	Evaluation Parameters	Results
8	Arash and Mahdi, 2013	Content	Single-Label	wiki-20 (5)	wiki-20 (20)	Concept Matching Based Approach (CMA)	Precision, Recall, F-Measure	Precision (0.61), Recall (0.58), F-Measure (0.60)
9	Hingmire et al, 2013	Content	Single-Label	20-Newsgroup (8), SRAA(10), WebKB (10)	20-Newsgroup, SRAA(73,218), WebKB (4,199)	Latent Dirichlet Allocation (LDA)	F-Measure	20-Newsgroup (0.92), SRAA(0.85), WebKB (0.71)
10	Wang and Sun, 2009	Content	Single-Label	Reuter (10), WebKB (7)	Reuter (21,578), WebKB (8,282)	NPE, Particle Swarm Optimization (PSO)	F-Measure	Reuter (0.94), WebKB (0.89)
11	Senthamarai and Ramaraj, 2008	Content	Single-Label	Not Reported	2,000 Documents	Particle Swarm Optimization (PSO)	Accuracy	Accuracy (0.90)
12	Guerrero et al., 2002	Content	Single-Label	7	202 Documents	Neural Network	Not Reported	Not Reported
13	Jingbo and Tianshun, 2002	Content	Single-Label	10	1000 Documents	Features Identification and Features Aggregation (FIFA)	Precision, Recall	Precision (0.80), Recall (0.76)

2.4.2 Analysis of Content Based Approaches (Multi-Label Classification)

The state-of-the-art approaches which exploit the content of documents and have performed multi-label classification are shown in the Table 2.2. These approaches have predicted one or more than one classes from the multiple classes [18, 54, 56, 58]. However, there are very few state-of-the-art approaches which perform multi-label classification. For multi-label classification, Santos [18] presented an approach which utilized ACM dataset and ACM classification system which contained 11 classes at its root level. They have applied different approaches (algorithms) on two different collections of documents (5000 and 10,000 documents) for the multi-label classification and achieved accuracy upto 0.88. Similarly, Lijuan [54] also performed multi-label classification and applied algorithm on different datasets like WIPO-alpha, Newsgroup and Enzyme etc and achieved accuracy upto 0.94 and precision upto 0.84.

2.4.3 Analysis of Metadata Based Approaches

The state-of-the-art approaches which exploit metadata of the documents have performed single-label classification. These approaches have been shown in the Table 2.3. These approaches predicted the most relevant class for a particular document from the multiple pre-defined classes. Very few state-of-the-art approaches have performed document classification by exploiting only metadata [13, 14, 35]. One important finding from literature is that the systems which utilize metadata of research papers were only able to classify papers into single class. For single-label classification, Flyn [35] has applied an algorithm on 2000 documents for the classification of documents into 99 pre-defined classes and achieved precision upto 0.79, recall value 0.81 and F-measure value 0.79. Khor [14] applied different algorithms on a collection of 400 documents but they used very few generic classes (i.e. 4 classes) and achieved accuracy upto 0.84 for their document classification technique.

In this dissertation, our focus is to classify scientific documents by using only metadata. There are very few state-of-the-art approaches that rely on freely available metadata as shown in Table 2.3. These schemes classify documents into single-label [13, 14, 35]. Only approaches proposed by Santos and Rodrigous [18], Wang and Desai [34] and Lijuan [54] classify documents to multiple classes but by exploiting content of the documents. All other approaches have not dealt with multi-label classification problem. The existing multi-label classification schemes classify documents into limited number of categories. Those researchers who have used metadata of the papers, they only perform the single-label classification and achieved up to almost 0.84 accuracy by using a few numbers of classes. The approach made by Santos and Rodrigues [18] is closely related to our experiments due to the following reasons: (1) They have utilized comprehensive dataset from different conferences, workshops, and journals, (2) Their proposed approach use multi-label classification, (3) they remained one of the important state-of-the-art approach which achieves the best accuracy and utilize the whole content, and (4) they have used all 11 categories on root level ACM hierarchy for classification. Others have experimented on limited number of categories. Furthermore, we want to see by using only the metadata how closely one can achieve in terms of accuracy? Therefore, we will be using their approach for comparisons.

Table 2.2: Critical Analysis of Content based Approaches for Multi-Label

S.No.	Approaches	Type of Data	Classification Type	No. of Classes	Dataset	Algorithm / Methodolgy	Evaluation Parameters	Results
1	Yan et al., 2018	Content	Multi-Label	Biomedicine (150), Email (6), News(103)	Biomedicine (100,000), Email (3, 021), News(800,000)	Long Short Term Memory (LSTM)	F-Measure	F-Measure (0.70)
2	Baker and Korhonen, 2017	Content	Multi-Label	PubMed (30)	PubMed (1,852)	INIT-A, INIT-B	Precision, Recall, F-Measure	Precision (0.73, 0.68), Recall (0.77, 0.83), F-Measure (0.75, 0.75)
3	Santos and Rodrigues, 2009	Content	Multi-Label	11	5,000 and 10,000 Documents	Binary Relevence, Naive Bayses Multi-Nomial, Multi-Label KNN	Accuracy	Accuracy (0.88)
4	Lijuan, 2008	Content	Multi-Label	WIPO-alpha (8), News-group (5), OHSUMED (15), EN-ZYM (236)	WIPO-alpha, Newsgroup (1,000), OHSUMED (54,708), EN-ZYM (9,455)	Hierarchical SVM, Hierarchical Perception	Accuracy, Precision	Accuracy (0.94), Precision (0.89)

Table 2.3: Critical Analysis of Metadata based Approaches

S.No.	Approaches	Type of Data	Classification Type	No. of Classes	Dataset	Algorithm / Methodolgy	Evaluation Parameters	Results
1	Flyn, 2014	Metadata	Single-Label	99	2,000 Documents	Independent Document Model (IDM) Framework	Precision, Recall, F-Measure	Precision (0.79), Recall (0.81), F-Measure (0.70)
2	Khor and Ting, 2006	Metadata	Single-Label	4	400 Documents	Baysian Network (BN), Naive Bayes (NB), Bayesian Network Learner (BNL)	Accuracy	BN (0.84), NB (0.83), BNL (0.76)

Chapter 3

Multi-Label Document Classification using Paper's Title and Keywords

This chapter introduces a framework to classify research articles into the multiple categories by employing the freely available metadata parameters. We are interested to scrutinize the potential of metadata based features by utilizing them in the best possible ways to assist in the scenarios when the content is not available. The primary observations and motivations that signify proposed framework are listed below:

1. The content based approaches attain more propitious results than metadata based approaches because of their richness in features, which leads towards the biasness of document classification community. But what should be done, when we do not have an access to the research articles' content? The major journal publishers like: ACM, Springer, Elsevier, IEEE etc. do not provide open access to their published articles as there are financial, legal and technical barriers. In such scenarios: there should be an alternative avenue to classify these articles. One best alternate is available in the form of freely available metadata. The metadata is defined as data about data or some external information about the actual data. Different kinds of useful research articles' metadata such as: title, authors, keywords, categories etc. are almost freely available in various digital libraries.

Table 3.1: Datasets Statistics

Features	**ACM** [18]	**J.UCS** [37]
Total Number of Research Papers	86,116	1,460
Total Number of Classes (Categories) at Root Level	11	13
Total Number of Categories' Levels	3	3
Total Number of Research Papers without Keywords' Section	3860	31
Single-Label Research Papers Percentage(%)	54%	51%
Multi-Label Research Papers Percentage(%)	46%	49%
Total Number of Journals or Conferences or Workshops	2,240	01

2. As per our knowledge, there exists no scheme in the literature that performs multi-label classification of scientific documents by relying fully on metadata; of course, it requires a comprehensive set of parameters. We argue that a research papers may belong to more than one category; the statement is validated via experiments on diversified datasets (see Table 3.2). There is a great possibility that a scientific document is partially associated with one class and partially related to other classes. For instance, a scientific document on "Similarity Algorithm for Gene Ontology Terms" has three associations: one with the "Genes (Biology)", second with the "Ontology", and third with "Similarity algorithms" class. The existing multi-label document classification techniques classify documents into a limited number of categories and their performance drops off when the number of categories gets increased.

The partial experimental results of this approach have been published by the author of this dissertation in a journal [36]. The proposed framework performs multi-label classification of scientific documents into pre-defined ACM subject hierarchy by using only metadata of the documents. The Title and Keywords of research papers are exploited in different combinations in this research. The system is intended to perceive the best metadata parameter at which the comprehensive system of research article classification can be formed.

3.1 Datasets

To comprehensively evaluate the proposed system, one needs to carefully select the dataset. To evaluate the proposed framework, we have carefully picked two best suited diversified datasets. One of them is based on research publications

Table 3.2: Datasets Statistics

Features	**ACM** [18]	**J.UCS** [37]
Total Number of Research Papers	86,116	1,460
Total Number of Classes (Categories) at Root Level	11	13
Total Number of Categories' Levels	3	3
Total Number of Research Papers without Keywords' Section	3860	31
Single-Label Research Papers Percentage(%)	54%	51%
Multi-Label Research Papers Percentage(%)	46%	49%
Total Number of Journals or Conferences or Workshops	2,240	01

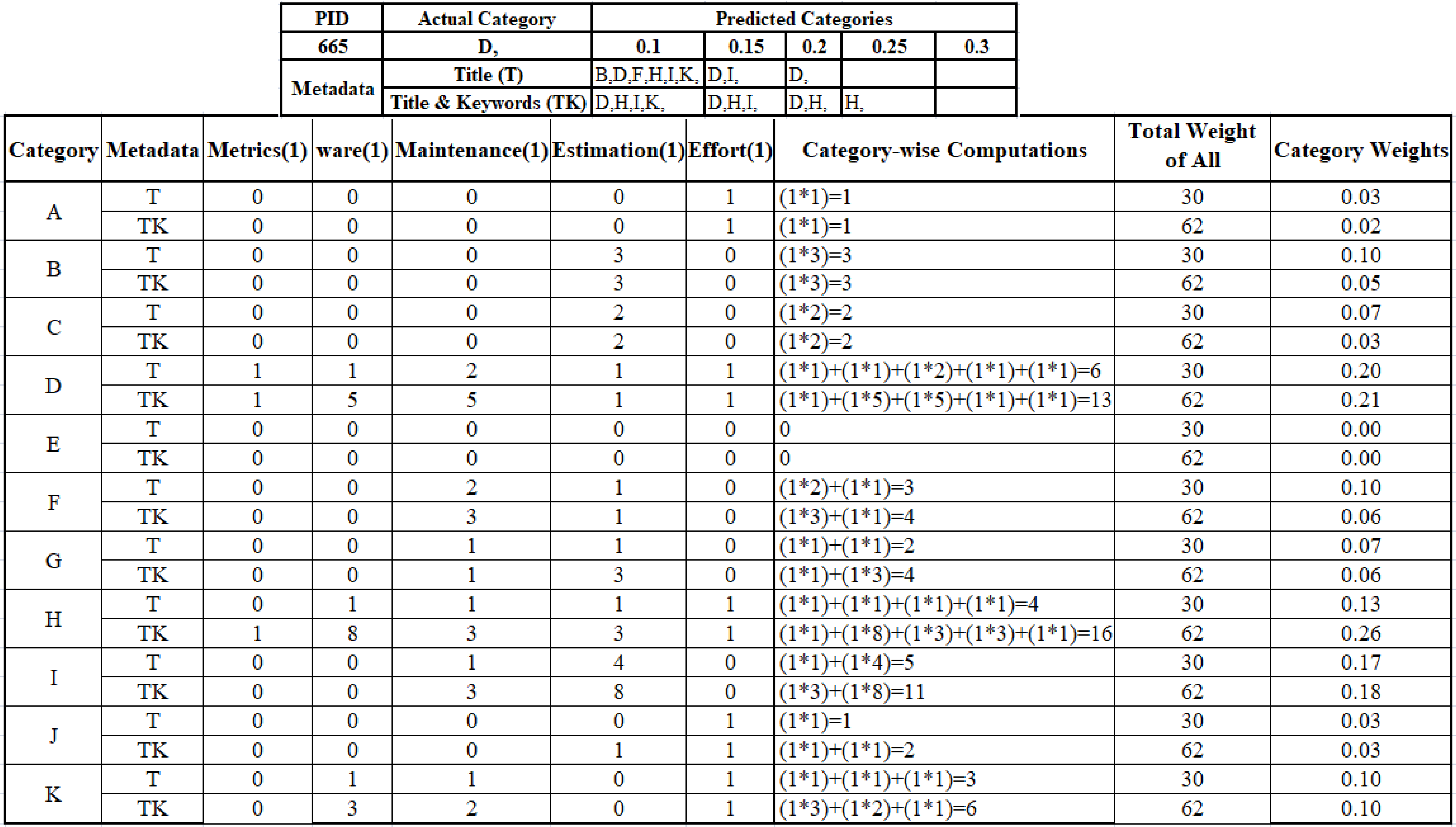
from Journal of Universal Computer Science (J.UCS) [37] and another one contains research publications from the Association of Computing Machinery (ACM) and developed by Santos *et al.* [18].The reason for the selection of J.UCS dataset is twofold: J.UCS covers all topics of Computer Science and the researchers who published their work belong to diversified domains and geographical regions, which can help us to perform comprehensive evaluation. The detailed statistics of both data sets are presented in Table 3.2. Similarly, the reason for the selection of ACM [18] dataset is that it contains research publications from different conferences, journals and the workshops. The detailed description of these two datasets is presented in the following sections.

J.UCS dataset contains 1,460 research publications. It has extended the ACM CCS98 with two more classes like L and M. Therefore, at top level, there are 13 distinct classes in J.UCS dataset rather than 11 classes as per ACM classification (i.e. classes A-K correspond to the ACM classification with its subclassifications, classes L (Science and Technology of Learning) and M (Knowledge Management) were added to reflect the development of the Computer Science discipline). However, ACM dataset built by Santos [18] contains 86,116 research publications from conferences, journals and workshops of diversified domains. Both datasets have significant numbers of research articles associated with multiple classes.

3.2 Multi-Label Document Classification Framework

This section describes a multi-label document classification mechanism in a formal way. The classification process involves two steps: 1) Training and 2) Testing. The number of documents along with their term-frequency obtained from Titles and Keywords and their belonging Categories are parsed for training phase.

Table 3.3: Test Document classification



PID	Actual Category	Predicted Categories				
665	D,	0.1	0.15	0.2	0.25	0.3
Metadata	Title (T)	B,D,F,H,I,K,	D,I,	D,		
	Title & Keywords (TK)	D,H,I,K,	D,H,I,	D,H,	H,	

Category	Metadata	Metrics(1)	ware(1)	Maintenance(1)	Estimation(1)	Effort(1)	Category-wise Computations	Total Weight of All	Category Weights
A	T	0	0	0	0	1	(1*1)=1	30	0.03
	TK	0	0	0	0	1	(1*1)=1	62	0.02
B	T	0	0	0	3	0	(1*3)=3	30	0.10
	TK	0	0	0	3	0	(1*3)=3	62	0.05
C	T	0	0	0	2	0	(1*2)=2	30	0.07
	TK	0	0	0	2	0	(1*2)=2	62	0.03
D	T	1	1	2	1	1	(1*1)+(1*1)+(1*2)+(1*1)+(1*1)=6	30	0.20
	TK	1	5	5	1	1	(1*1)+(1*5)+(1*5)+(1*1)+(1*1)=13	62	0.21
E	T	0	0	0	0	0	0	30	0.00
	TK	0	0	0	0	0	0	62	0.00
F	T	0	0	2	1	0	(1*2)+(1*1)=3	30	0.10
	TK	0	0	3	1	0	(1*3)+(1*1)=4	62	0.06
G	T	0	0	1	1	0	(1*1)+(1*1)=2	30	0.07
	TK	0	0	1	3	0	(1*1)+(1*3)=4	62	0.06
H	T	0	1	1	1	1	(1*1)+(1*1)+(1*1)+(1*1)=4	30	0.13
	TK	1	8	3	3	1	(1*1)+(1*8)+(1*3)+(1*3)+(1*1)=16	62	0.26
I	T	0	0	1	4	0	(1*1)+(1*4)=5	30	0.17
	TK	0	0	3	8	0	(1*3)+(1*8)=11	62	0.18
J	T	0	0	0	0	1	(1*1)=1	30	0.03
	TK	0	0	0	1	1	(1*1)+(1*1)=2	62	0.03
K	T	0	1	1	0	1	(1*1)+(1*1)+(1*1)=3	30	0.10
	TK	0	3	2	0	1	(1*3)+(1*2)+(1*1)=6	62	0.10

To understand the working of the proposed framework, one needs to understand the Table 3.3. In Table 3.3, first column represent the total number of categories present in the dataset and these categories contain different number of documents. The metadata columns represent the type of metadata from where we have extracted the terms like either from the Titles or Keywords or from both Titles & Keywords, in this Table 3.3 we have used Title and Titles & Keywords. Next columns represent the number of terms and their frequencies contained by the document that will be matched with category-wise terms which results in category-wise computations. This computation is presented in category-wise computation column. In Table 3.3, all the non-zero values for all categories mean that all the categories contain particular words of the test document with that frequency. Frequency means that how many times that particular word is repeated in the particular category. Non-zero value is actually representing the frequency of that word in Titles' string or Title & Keywords' string a category. Zero value (frequency) means that word in a test document is not exist in that particular category. We have computed the category-wise weight for the document by multiplying the both frequencies of matched document's term with category's term. Similarly, adding all the weights of matched document's terms. If no term of the document matched with the terms of category then its weight becomes zero. For instance, in Table 3.3, we can notice that for category E, there is no document's terms are matched so its weight is zero. We have also computed the total weight of all categories for each metadata.

For instance, in Table 3.3, we have computed the total weight of 30 for metadata (*Title*) and 62 for metadata (*Title* & *Keywords*). In the next column, we have computed the contribution scores (membership value) of the document in each category according to the metadata. For instance, we have computed the membership values of document for category A which are 0.03 for metadata (*Title*) and 0.02 for metadata (*Title* & *Keywords*). After calculating these membership values for each category, the next step is to predict the most suitable category for that document. For this we have applied different threshold values (as shown in Table 3.3) for the prediction of the categories. In this way, we have trained our system (proposed approach) for the multi-label classification of the documents.

Subsequently, when a test document is received as an input to be classified, the proposed approach tokenizes the Title and Keywords of the document and identify the contribution scores of each term in the training dataset' classes. Based on the maximum score found in a class, the class is predicted for the test document based on some threshold value which will be discussed in detail in the Section 3.3.2. In this way, more than one class can be predicted, as the terms could be matched with more than one class depending upon the training dataset's classes score contributions. To decide what value of score belongs to which class, the threshold value is defined by conducting multiple experimental rounds in Section 3.3.2. Defining the most optimal threshold value is a challenging task, for instance, see Table 3.3, we have a test document with PID 665, it contains terms like metrics, ware, maintenance, estimation and effort having frequency 1 for all. We have computed the contribution score of the document with respect to each class and predicted the number of classes. On the basis of contribution score, class for the input article is predicted by following the certain threshold criteria. At the threshold value 0.1, there are large numbers of predicted classes as compared to the original class (D) of the document. As the value of threshold increases, the ratio of the predicted values decreases and it becomes zero at threshold value 0.3.

The framework of the proposed system is presented in Figure 3.1. The research

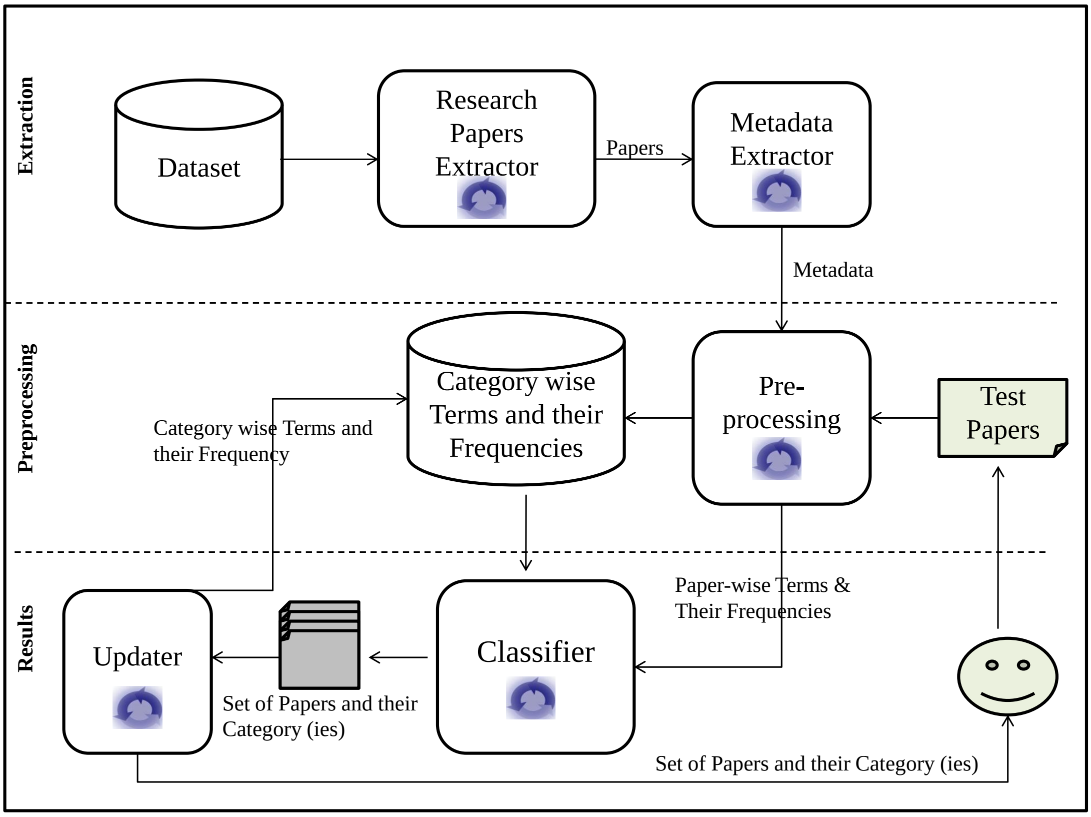

Extraction
Dataset
Research Papers Extractor
Papers
Metadata Extractor
Metadata
Preprocessing
Category wise Terms and their Frequency
Category wise Terms and their Frequencies
Pre-processing
Test Papers
Results
Updater
Set of Papers and their Category (ies)
Classifier
Paper-wise Terms & Their Frequencies
Set of Papers and their Category (ies)

Figure 3.1: Multi-Label Document Classification Framework

paper extractor module extracts reseach papers from the dataset.The system takes these research papers as an input to the metadata extractor module. The metadata extractor extracts metadata like Title, Keywords and Categories of research papers from it and send to the preprocessor module as an input. After applying some pre-processing steps, these metadata parameters are parsed and send to *Category wise Metadata Merger (CMM)* algorithm to merge category wise metadata of each research paper and stored category wise terms and their frequencies in a database. The Multi-Label Classifier (MLC) predicts these research papers into one or more than one categories. To enhance the performance of our approach, the category updater is utilized to enrich our knowledge base (dataset) for the articles' classification.

The experiments on diversified datasets show that there are significant numbers of articles that pertain to multiple categories as shown in Table 3.2. There is a great possibility that a research paper is partially associated with one category and is partially related to other categories. We have developed a mechanism for the identification of such type of multiple classes. To tackle such type of overlapping, first the membership (here we find terms frequency weights or simply we can say weight or association for each category) of scientific documents is found with respect to each category and then the alpha-cut "φ" (threshold) is applied on that membership to identify the most relevant set of categories for the scientific documents. The formal representation of identifying categories for scientific documents is in Eq. 3.1:

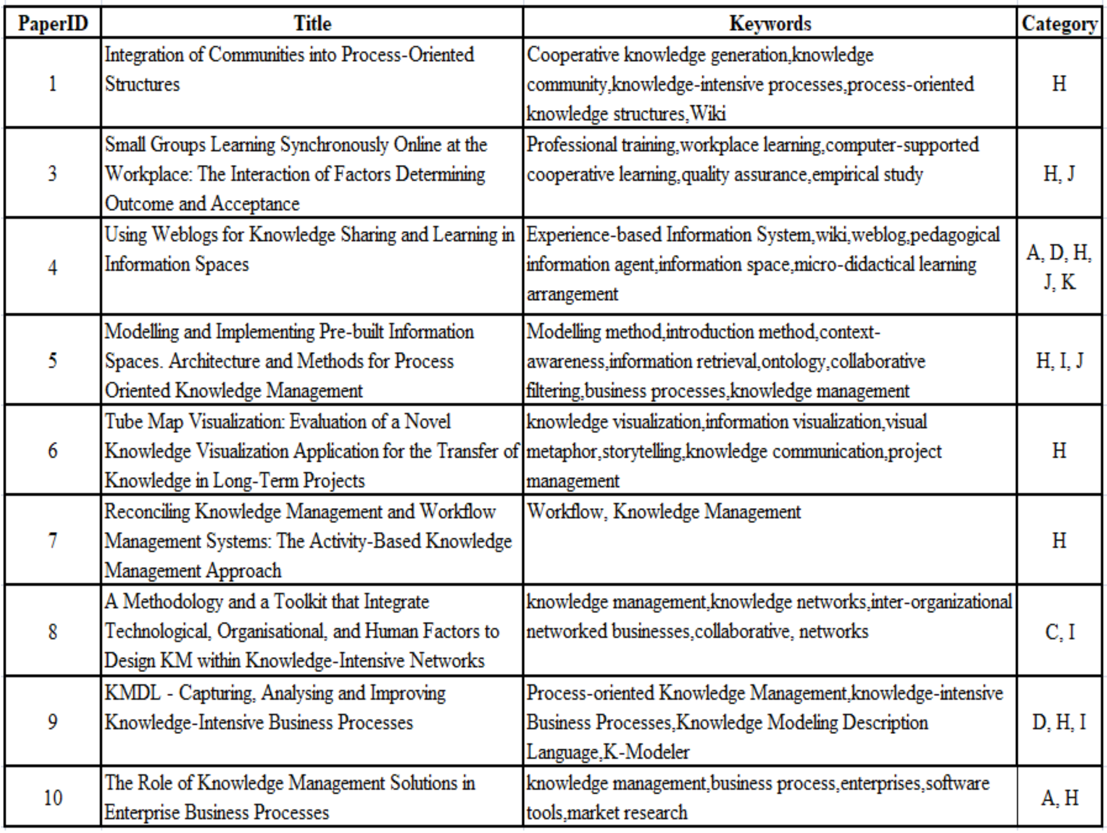
$$\forall \mu C_i(\mathfrak{D}) \geq \varphi \implies \mathfrak{D} : C_i \tag{3.1}$$

Where $\mathfrak{D}$ is the scientific document, C_i is the set of categories, φ is an alpha-cut (threshold) which can be assign any value determine by domain experts; $\mu C_i(\mathfrak{D})$ is the membership (terms frequency weights) of $\mathfrak{D}$ in category C_i and $\mathfrak{D} : C_i$ represents that scientific document $\mathfrak{D}$ belongs to category set C_i.

3.2.1 Pre-processing

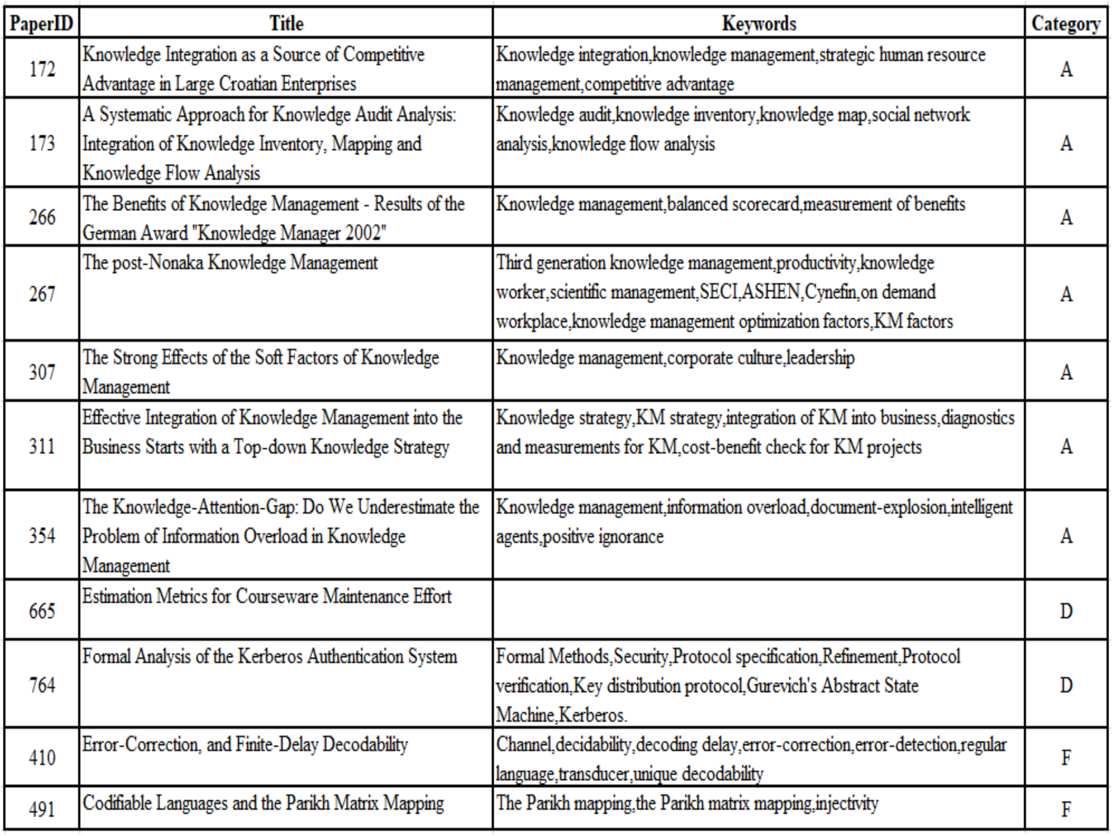
For the proposed framework, the metadata based features are extracted from the research publications of the J.UCS dataset. From this dataset, three relational tables are extracted which are used for feature selection process. These relations are shown in the Figure 3.2. The articles relation can assist to acquire important

Table 3.4: Research Papers' Category Pairs (Sample)



PaperID	Title	Keywords	Category
1	Integration of Communities into Process-Oriented Structures	Cooperative knowledge generation,knowledge community,knowledge-intensive processes,process-oriented knowledge structures,Wiki	H
3	Small Groups Learning Synchronously Online at the Workplace: The Interaction of Factors Determining Outcome and Acceptance	Professional training,workplace learning,computer-supported cooperative learning,quality assurance,empirical study	H, J
4	Using Weblogs for Knowledge Sharing and Learning in Information Spaces	Experience-based Information System,wiki,weblog,pedagogical information agent,information space,micro-didactical learning arrangement	A, D, H, J, K
5	Modelling and Implementing Pre-built Information Spaces. Architecture and Methods for Process Oriented Knowledge Management	Modelling method,introduction method,context-awareness,information retrieval,ontology,collaborative filtering,business processes,knowledge management	H, I, J
6	Tube Map Visualization: Evaluation of a Novel Knowledge Visualization Application for the Transfer of Knowledge in Long-Term Projects	knowledge visualization,information visualization,visual metaphor,storytelling,knowledge communication,project management	H
7	Reconciling Knowledge Management and Workflow Management Systems: The Activity-Based Knowledge Management Approach	Workflow, Knowledge Management	H
8	A Methodology and a Toolkit that Integrate Technological, Organisational, and Human Factors to Design KM within Knowledge-Intensive Networks	knowledge management,knowledge networks,inter-organizational networked businesses,collaborative, networks	C, I
9	KMDL - Capturing, Analysing and Improving Knowledge-Intensive Business Processes	Process-oriented Knowledge Management,knowledge-intensive Business Processes,Knowledge Modeling Description Language,K-Modeler	D, H, I
10	The Role of Knowledge Management Solutions in Enterprise Business Processes	knowledge management,business process,enterprises,software tools,market research	A, H

Table 3.5: Research Papers of Same Category (Sample)



PaperID	Title	Keywords	Category
172	Knowledge Integration as a Source of Competitive Advantage in Large Croatian Enterprises	Knowledge integration,knowledge management,strategic human resource management,competitive advantage	A
173	A Systematic Approach for Knowledge Audit Analysis: Integration of Knowledge Inventory, Mapping and Knowledge Flow Analysis	Knowledge audit,knowledge inventory,knowledge map,social network analysis,knowledge flow analysis	A
266	The Benefits of Knowledge Management - Results of the German Award "Knowledge Manager 2002"	Knowledge management,balanced scorecard,measurement of benefits	A
267	The post-Nonaka Knowledge Management	Third generation knowledge management,productivity,knowledge worker,scientific management,SECI,ASHEN,Cynefin,on demand workplace,knowledge management optimization factors,KM factors	A
307	The Strong Effects of the Soft Factors of Knowledge Management	Knowledge management,corporate culture,leadership	A
311	Effective Integration of Knowledge Management into the Business Starts with a Top-down Knowledge Strategy	Knowledge strategy,KM strategy,integration of KM into business,diagnostics and measurements for KM,cost-benefit check for KM projects	A
354	The Knowledge-Attention-Gap: Do We Underestimate the Problem of Information Overload in Knowledge Management	Knowledge management,information overload,document-explosion,intelligent agents,positive ignorance	A
665	Estimation Metrics for Courseware Maintenance Effort		D
764	Formal Analysis of the Kerberos Authentication System	Formal Methods,Security,Protocol specification,Refinement,Protocol verification,Key distribution protocol,Gurevich's Abstract State Machine,Kerberos.	D
410	Error-Correction, and Finite-Delay Decodability	Channel,decidability,decoding delay,error-correction,error-detection,regular language,transducer,unique decodability	F
491	Codifiable Languages and the Parikh Matrix Mapping	The Parikh mapping,the Parikh matrix mapping,injectivity	F

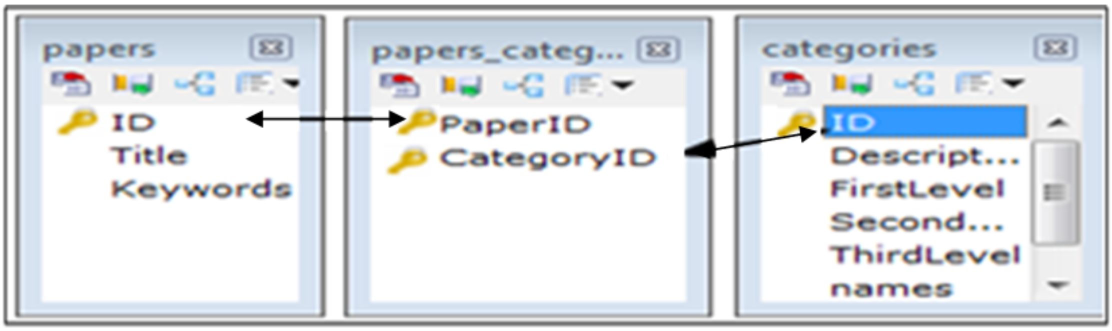


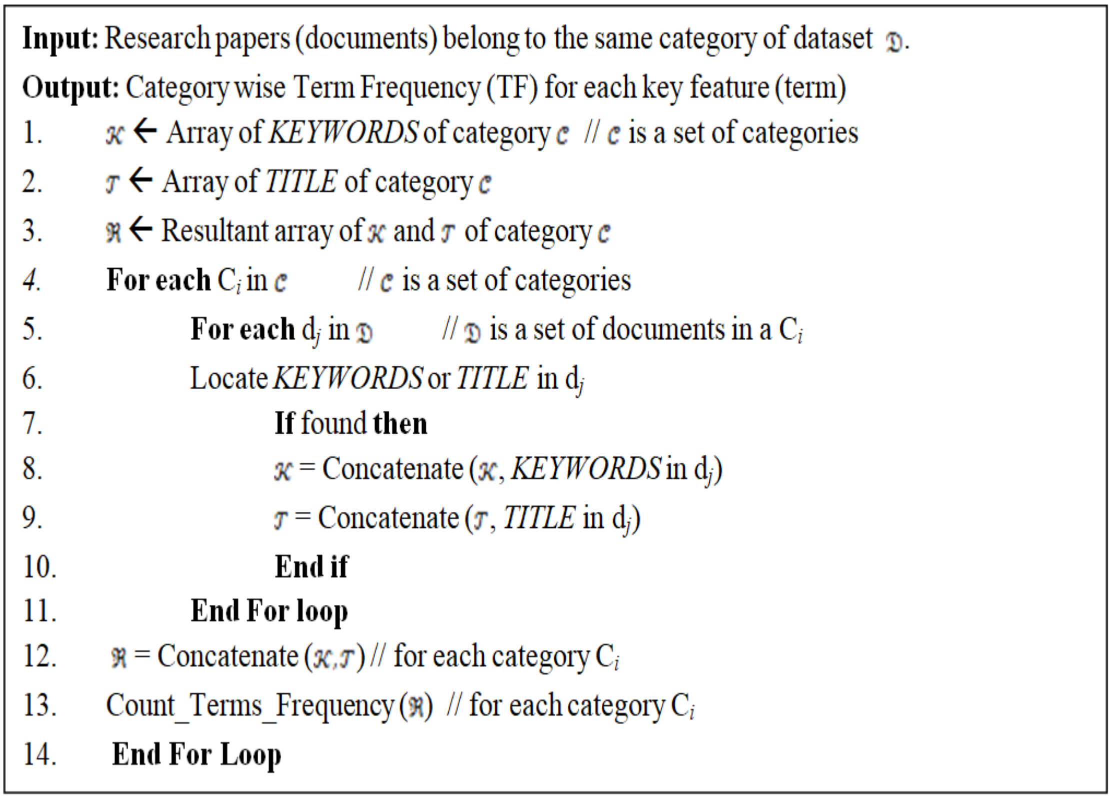
FIGURE 3.2: J.UCS dataset's Tables

metadata parameters such as research paper's Title, Keywords etc. The category's relation has information about the classification system used by the J.UCS to assign relevant categories to the research papers. It contains information like category's label, it's different levels etc. The third relation is basically the relationship between the Papers and Category, which depicts that the article belongs to which category. The research papers' category pairs are generated by using these three relations from the J.UCS dataset.
The papers' category pairs contain *Title*, *Keywords* and original annotated categories of the research papers. The sample of papers' categories pairs is shown in the Table 3.4. In the Table 3.4, it can be seen that some papers belong to more than one category, which motivates our multi-label classification further. To make metadata parameters ready for experiments, some pre-processing steps like normalization (conversion of all words into lowercase), removal of stop words from the *Title* and *Keywords* of all research papers and conversion of all compound words into the single words are performed.

3.2.2 Category-wise Metadata Merger Algorithm

Category-wise Metadata Merger (CMM) Algorithm merges the metadata of all research papers that belong to the same category. In Table 3.5, we can see that research paper such as 172, 173, 266, 267, 307, 311 and 354 are belong to the category A. Similarly, research papers 665 & 764 are belong to category D and research papers 410 & 491 are belong to category F. The CMM algorithm is presented in the Figure 3.3. The CMM algorithm extracts already processed *Title* and *Keywords* from each research paper and merges (concatenates) them according to their categories. The *Titles* of all research papers relating to each category are separately concatenated and the term frequency (TF) weights for these *Titles* strings are computed. Similarly, *Keywords* of all research papers relating to each category are concatenated and the term frequency (TF) weights for these *Keywords* strings are computed. Both, the *Titles* & *Keywords* are collectively concatenated for all research papers relating to each category and the term frequency (TF) weights for these *Titles* & *Keywords* string are computed. These TF weights with respect to *Title*, *Keywords* and *Title* & *Keywords* strings are stored in the database with respect to their categories and are computed by using the algorithm presented in the Figure 3.3.The algorithm works as follows:

In pre-processing steps, the research papers according to their already annotated categories are stored in the database. These categories-wise research papers are given as input to this algorithm and it returns category-wise TF weights. In the

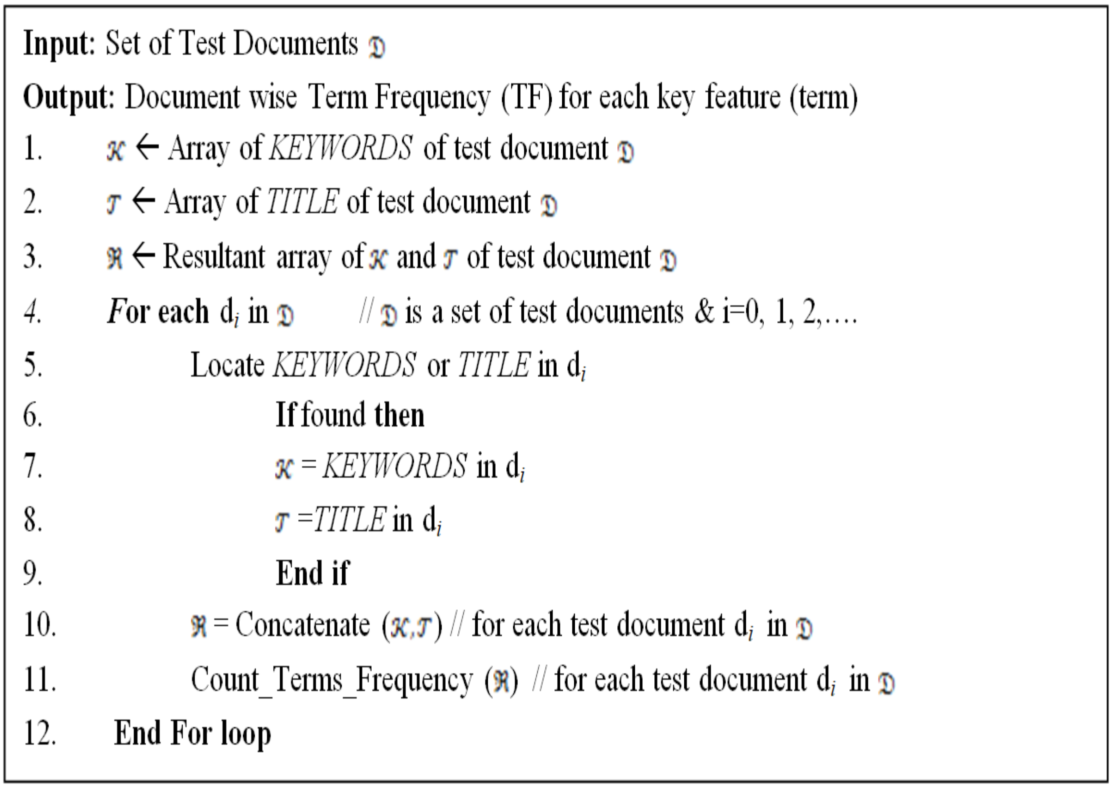


```
Input: Research papers (documents) belong to the same category of dataset $\mathfrak{D}$.
Output: Category wise Term Frequency (TF) for each key feature (term)
1.    $\mathcal{K}$ ← Array of KEYWORDS of category $\mathcal{C}$  // $\mathcal{C}$ is a set of categories
2.    $\mathcal{T}$ ← Array of TITLE of category $\mathcal{C}$
3.    $\Re$ ← Resultant array of $\mathcal{K}$ and $\mathcal{T}$ of category $\mathcal{C}$
4.    For each $C_i$ in $\mathcal{C}$          // $\mathcal{C}$ is a set of categories
5.        For each $d_j$ in $\mathfrak{D}$          // $\mathfrak{D}$ is a set of documents in a $C_i$
6.        Locate KEYWORDS or TITLE in $d_j$
7.            If found then
8.            $\mathcal{K}$ = Concatenate ($\mathcal{K}$, KEYWORDS in $d_j$)
9.            $\mathcal{T}$ = Concatenate ($\mathcal{T}$, TITLE in $d_j$)
10.           End if
11.       End For loop
12.   $\Re$ = Concatenate ($\mathcal{K}$,$\mathcal{T}$) // for each category $C_i$
13.   Count_Terms_Frequency ($\Re$)  // for each category $C_i$
14.   End For Loop
```

Figure 3.3: Category wise Metadata Merger Algorithm

Figure 3.3, the algorithm picks research paper one by one from each category (at Line 4 & 5) and extracts *Title* of each research paper in a category and concatenates all *Titles* (at Line 9) of that particular category. Similarly, it extracts *Keywords* of each research paper (if exist) in a category and concatenates all *Keywords* (at Line 8) of that particular category. At the end of Line 10, we have a resultant strings of *Title* and *Keywords* for each each category. Then these resultant strings

of *Title* and *Keywords* are concatenated for each category (at Line 12) and these categories-wise resultant strings are given as inputs to the Count_Terms_Frequency module (at Line 13) to compute TF weights for each category. At the end of Line 14, we have computed category wise of all terms and their frequecies.
Computing Term Frequency Weights of Test Document, algorithm is proposed to compute test document's term frequency weights. The working of this algorithm is depicted in the Figure 3.4. When a user inputs a research paper (test document) for identification of its set of categories, the same pre-processing steps are adopted as discussed above. From the *Title* and *Keywords* of the research paper, the term frequency weights are computed for each term of the research article. This algorithm takes test document as input and returns TF against each test documents. It extracts *Title* and *Keywords* from the test documents and then concatenates *Title* & *Keywords* to form the resultant string. Then the resultant string is given to the Count_Terms_Frequency module to compute TF weights for that test document. These TF weights return as an output to the user.



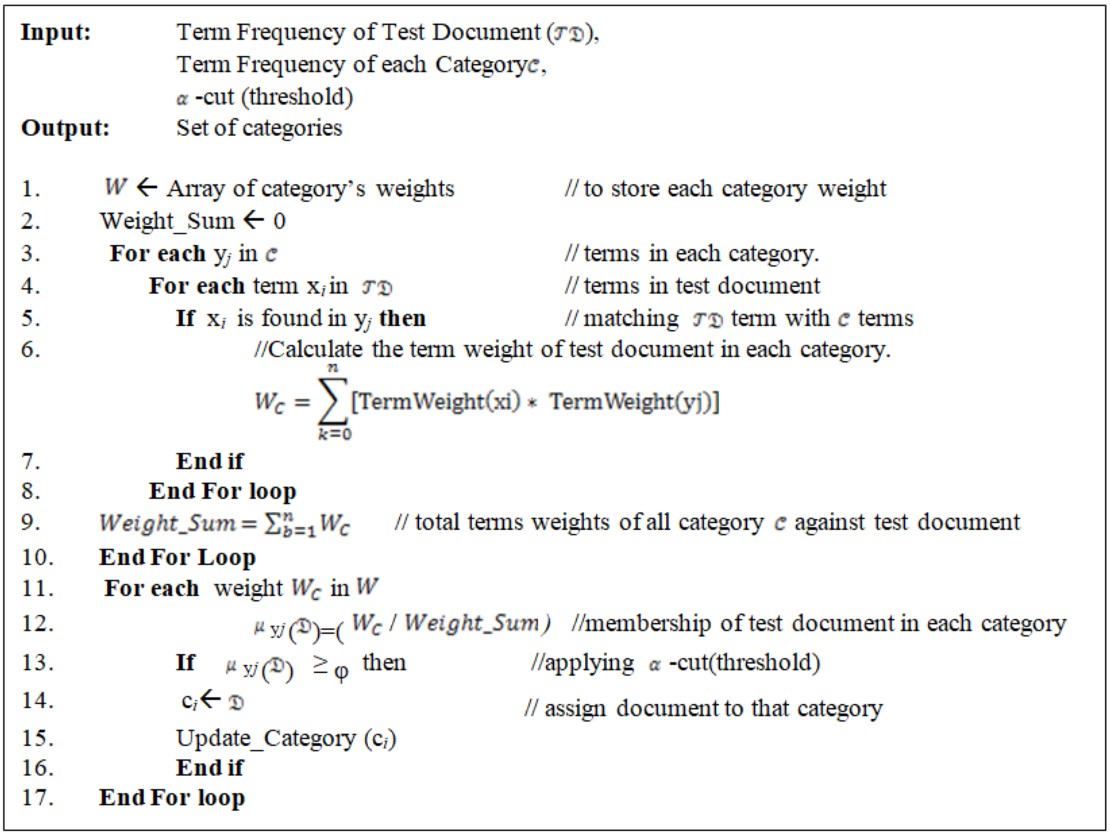
```
Input: Set of Test Documents $\mathfrak{D}$
Output: Document wise Term Frequency (TF) for each key feature (term)
    $\mathcal{K}$ ← Array of KEYWORDS of test document $\mathfrak{D}$
    $\mathcal{T}$ ← Array of TITLE of test document $\mathfrak{D}$
    $\mathfrak{R}$ ← Resultant array of $\mathcal{K}$ and $\mathcal{T}$ of test document $\mathfrak{D}$
    For each $d_i$ in $\mathfrak{D}$        // $\mathfrak{D}$ is a set of test documents & i=0, 1, 2,....
        Locate KEYWORDS or TITLE in $d_i$
            If found then
            $\mathcal{K}$ = KEYWORDS in $d_i$
            $\mathcal{T}$ =TITLE in $d_i$
            End if
        $\mathfrak{R}$ = Concatenate ($\mathcal{K}$,$\mathcal{T}$) // for each test document $d_i$ in $\mathfrak{D}$
        Count_Terms_Frequency ($\mathfrak{R}$) // for each test document $d_i$ in $\mathfrak{D}$
    End For loop
```

FIGURE 3.4: Algorithm for Computing Term Frequency Weights of Test Document

3.2.3 Multi-Label Classification (MLC)

The MLC is the core algorithm of proposed framework. It helps a user to find-out the most relevant category or class for the input research article. When a user inputs a test document for the identification of its set of categories. To compute TF weights from the metadata of the test documents like *Title* and *Keywords*, we used the algorithm which is presented in the Figure 3.4. Similarly, we have already computed category-wise TF weights by using the CMM algorithm. There are three inputs for the MLC algorithm; first is the test document's terms and their frequencies, second input is the category wise terms and their frequencies and third input the threshold value which is selected by the domain expert. The next task is to compare the test documents TF weights with all categories-wise TF weights. For this comparison, the test document TF weights and category-wise TF weights are given as an input to the Multi-Label Classification (MLC) algorithm and it returns set of categories as output of the MLC algorithm. After the comparison of these TF weights, the weights (W_c) for each category is calculated from Line 3 to Line 8 (the comparison is also presented above in the Table 3.3). The total weight ($Weight\_Sum$) of all the categories is also computed (at Line 9). The next task is to compute membership ($\mu y_j(\mathfrak{D})$) (some weights or association of test document with each category) of the test document with respect to each category (at Line 11 & 12). At the end, it is predicted that how to assign the most relevant category to the test document (at Line 13 & 14). For this purpose, MLC predicts the most relevant category or set of categories for the test document on the basis of its membership in each category.

The higher membership increases the chances of assigning that category to the test documents. At this point, there are two possibilities to predict the most relevant category or categories for the test document. One is to select the top most category which has a higher membership among all categories and another possibility is by applying some "φ" α-cut (threshold) to predict one or more than one category. This threshold value is selected by the domain expert (which is discussed in results section 3.3.2). After the prediction of category or set of categories for the test document, the next task is to update the knowledge-base (repository) of the proposed framework (at Line 15). For this, MLC updates terms frequency weights for the particular category or set of categories acquired against the test document. In this way, the knowledge-base is enriched for document classification to enhance the performance of our classification approach.



```
Input:    Term Frequency of Test Document ($\mathcal{T}\mathfrak{D}$),
          Term Frequency of each Category$c$,
          $\alpha$ -cut (threshold)
Output:   Set of categories

$W$ ← Array of category's weights          // to store each category weight
Weight_Sum ← 0
 For each $y_j$ in $c$                     // terms in each category.
     For each term $x_i$ in $\mathcal{T}\mathfrak{D}$              // terms in test document
        If $x_i$ is found in $y_j$ then      // matching $\mathcal{T}\mathfrak{D}$ term with $c$ terms
              //Calculate the term weight of test document in each category.
              $W_C = \sum_{k=0}^{n}$ [TermWeight(xi) * TermWeight(yj)]
        End if
     End For loop
    $Weight\_Sum = \sum_{b=1}^{n} W_C$     // total terms weights of all category $c$ against test document
End For Loop
 For each weight $W_C$ in $W$
              $\mu_{yj}(\mathfrak{D})$=( $W_C$ / $Weight\_Sum$ )  //membership of test document in each category
        If $\mu_{yj}(\mathfrak{D}) \geq \varphi$ then        //applying $\alpha$ -cut(threshold)
         $c_i$ ← $\mathfrak{D}$                     // assign document to that category
        Update_Category ($c_i$)
        End if
End For loop
```

Figure 3.5: Multi-Label Classification (MLC) Algorithm

3.3 Results and Analysis

The proposed framework is implemented and tested on two diversified datasets, one is the Journal of Universal Computer Science (J.UCS) dataset [37] and another is ACM dataset developed by Santos *et al.* [18]. The reasons for the selection of J.UCS dataset are twofold: 1) the J.UCS covers all areas of Computer Science topics, 2) the authors belong to diversified domains which would help us in the comprehensive evaluation of the proposed approach. Similarly, the reason for the selection of ACM [18] dataset is that it contains research publications from the different conferences, journals and the workshops. The detailed description of these two datasets is presented in the section 3.3.3 and section 3.3.4 respectively. The results of these datasets are evaluated on the following evaluation parameters.

3.3.1 Evaluation Parameters

We performed comprehensive experiments on different and diversified datasets and evaluated the results of these experiments by applying well-known evaluation measures for the multi-label classification. These evaluation parameters are accuracy, precision, recall and F-measure and formulas for these evaluation parameters for multi-label document classification proposed by Godbole and Sarawagi [62]. These formulas are described below:

Accuracy: For each instance (research article), proportion of the predicted correct categories to the total number of (distinct actual and predicted) categories for that research paper. Average Accuracy can be computed by using the following formula which is shown in Eq. 3.2. Where Predicted categories are denoted as Pr, Actual categories are denoted as Ac and n is the total number of papers.

$$Accuracy{=}\frac{1}{n}\sum_{i=1}^{n}\frac{|Pr_i \bigcap Ac_i|}{|Pr_i \bigcup Ac_i|} \tag{3.2}$$

Precision: For each instance (research article), proportion of the predicted correct categories to the total number of predicted categories for that research paper. Average Precision can be computed by using the following formula which is shown in Eq. 3.3.

$$Precision{=}\frac{1}{n}\sum_{i=1}^{n}\frac{|Pr_i \bigcap Ac_i|}{|Pr_i|} \tag{3.3}$$

Recall: For each instance (research article), proportion of the predicted correct categories to the total number of actual categories for that research paper. Average Recall can be computed by using the following formula which is shown in Eq. 3.4.

$$Recall{=}\frac{1}{n}\sum_{i=1}^{n}\frac{|Pr_i \bigcap Ac_i|}{|Ac_i|} \tag{3.4}$$

F-Measure: It can be calculated by using the following formula which is shown in Eq. 3.5.

$$F\text{-}Measure{=}\frac{1}{n}\sum_{i=1}^{n}\left(\frac{2(Precision_i)(Recall_i)}{Precision_i + Recall_i}\right) \tag{3.5}$$

3.3.2 Threshold (α-cut) Tuning

For the extrapolation of one or more than one categories, the proposed MLC algorithm (see Figure 3.4) requires a threshold value which acts as a barrier for the prediction of categories. The threshold value is selected from the membership ($\mu y_j(\mathfrak{D})$) values computed by the MLC algorithm. For a test paper, if the computed membership value for a category is equal or higher than threshold value, then category is included in the set of predicted categories for a test paper.
Now the question crops up that how the optimal threshold value could be selected by domain experts? The comprehensive series of experimental rounds is performed to select the optimal threshold value. The first experiment is performed on a dataset comprising of 100 research papers to select a range of values on which a significant improvement in results could be seen by computing evaluation parameter values at each threshold value and used as the basis for comparing results acquired on different thresholds. For this purpose, the evaluation process is started from a minimum threshold value, for instance 0.05 and evaluated the performance of proposed technique. Similarly, the threshold value has gradually increased from 0.05 to 0.4 and so on. After in-depth analysis of results, it is analyzed that good results are achieved at a threshold value ranges from 0.1 to 0.3. But at the threshold value less than 0.1 (< 0.1), too many categories are predicted. For instance, if a research paper has one original category, the MLC algorithm predicted 5 categories at 0.1 threshold value. Similarly, there are multiple cases in which predicted categories are more than the original categories at threshold value less than 0.1(< 0.1). The prediction of a large number of categories is reported due to very low value of threshold value. On the other hand, when the threshold value is set to more than 3.0 (> 3.0), the numbers of predicted categories became very low or even there are many cases where the MLC algorithm predicted no category against the research paper due to the high value of threshold. For example, if a research paper has 5 original categories and at threshold value 3.0, in the most of cases the proposed model predicted only one category.
After the selection of the range of values for threshold, the next task is to acquire a suitable threshold value which could actually be used by the proposed algorithm. For this, we have performed experiments on a dataset comprising of 100 research papers randomly selected from J.UCS and ACM. These 100 research papers are taken from 11 different ACM categories (topics); for instance, “Information systems”, “ Computer Systems Organization”, “Theory of Computing”

etc. The evaluation parameters like precision, recall, F-measure and accuracy are computed for different data sizes of research papers like 10, 20, 30,. . . , 100 at different threshold values like 0.10, 0.11, 0.12,. . . , 0.30. After the detailed analysis of the first type of experiments, the most suitable threshold values are identified at which the MLC algorithm yields significant results. In the second type of experiments, the best threshold value of the first type of experiments is evaluated on 12 times bigger data size of research papers. Finally, the best threshold value of the first type of experiments is evaluated on even larger data size i.e. 15 times of the first experiments. These both bigger datasets belong to two different datasets such as: J. UCS [37] and ACM [18]. These three types of experiments helped us to find out the most optimum threshold value.

For the first type of experiments, we have utilized metadata (*Title* & *Keywords*) of first 50 research papers from each J.UCS and ACM dataset (Total 100 research papers). Some of these papers belong to only one category (Single-label) but most of these papers have more than one category (Multi-label). After the pre-processing steps, the proposed MLC algorithm is applied for the multi-label document classification. Our aim of these first types of experiments is to identify the most suitable threshold value at which the proposed MLC algorithm yields significant results. The results of these 100 research papers are evaluated as follows: the evaluation parameter values for 10 research papers are computed and the threshold value of 0.15 is identified at which MLC algorithm yields the best results. In the next experiment, 10 more research papers are added to validate the previous best threshold value. When we analyzed the results for 20 research papers, we came to know that the best threshold value is 0.17. Similarly, we have incremented our data size by adding 10 more research papers to validate the previous threshold values. We have noticed that the best threshold value is increasing gradually by increasing the number of research papers. When we increase the number of papers, at 70, we achieved the threshold value of 0.2 which remained constant or negligible change has been occurred while increasing the number of papers from 70 to 100. Each set of research papers is evaluated on threshold range from 0.1 to 0.3 by using the above evaluation parameters and analyzed the best threshold value for each set of research papers at which we have achieved the best accuracy. In the Figure 3.6, we have shown each set of research papers on X-axis and on the Y-axis, the only best threshold values is presented on which we achieved the best accuracy for each set of research papers. Hence from the first type of experiments on 100 research papers, the proposed model achieved best results at threshold value of 0.2.

In the above experiments for 100 research papers, the Term Frequencies (TF) weights for both *Title* & *Keywords* are computed by using the algorithm as depicted in the Figure 3.4. The category-wise TF weights for metadata (*Title*), metadata (*Keywords*) and for both metadata (*Title* & *Keywords*) are already stored separately. Then these research papers and their TF weights are given to the MLC classifier for the prediction of set of categories for these research articles. The exact matching has been performed between metadata (*Title* & *Keywords*) of research paper's terms and stored category-wise terms for metadata (*Title* & *Keywords*) and computed the membership ($\mu y_j(\mathfrak{D})$) values for each categories. We come to know that membership values are different for each category. These membership values for each category are dependent on the frequent occurrence of the test paper's terms in each category. We have applied different threshold values like 0.1, 0.11, 0.12, . . . , 0.3 which are selected from these membership values to predict the set of categories for article.

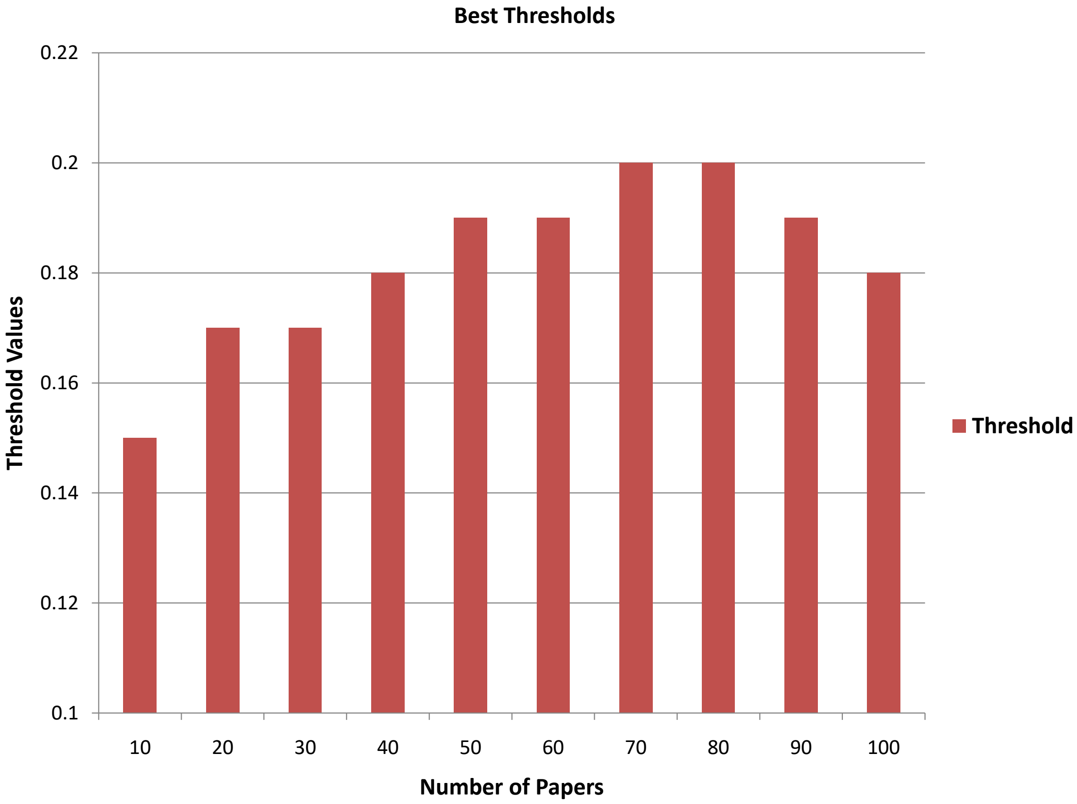


Figure 3.6: Number of Papers Vs Threshold Values

The MLC algorithm has predicted set of categories for 100 research papers and these predicted categories are evaluated by utilizing the well-known evaluation measures as described above in Section 3.3.1. Each set of research papers is evaluated on threshold range from 0.1 to 0.3 and analyzed the best threshold value for each set of research papers at which we have achieved the best accuracy. Hence from the first type of experiments on 100 research papers our proposed model achieved best results at threshold value of 0.2 and the results of all the best threshold values on which we achieved the best results for each 10 set of research papers are presented in the Figure 3.7 and we have plotted evaluation parameters on Y-axis and best threshold values (membership values) on X-axis. The best results for evaluation parameters such as accuracy, precision and F-measure are attained at threshold value of 0.2. The best result for recall is achieved at threshold value of 0.1.

From above first type of experiments, it is found that the optimum threshold values is 0.2 and we have further investigated this predicted threshold value (0.2) with more experiments on two different larger and diversified datasets i.e. 12 and 15 times bigger for the validation of this optimum threshold value. This threshold value is considered as a benchmark for the prediction of categories by the MLC algorithm.

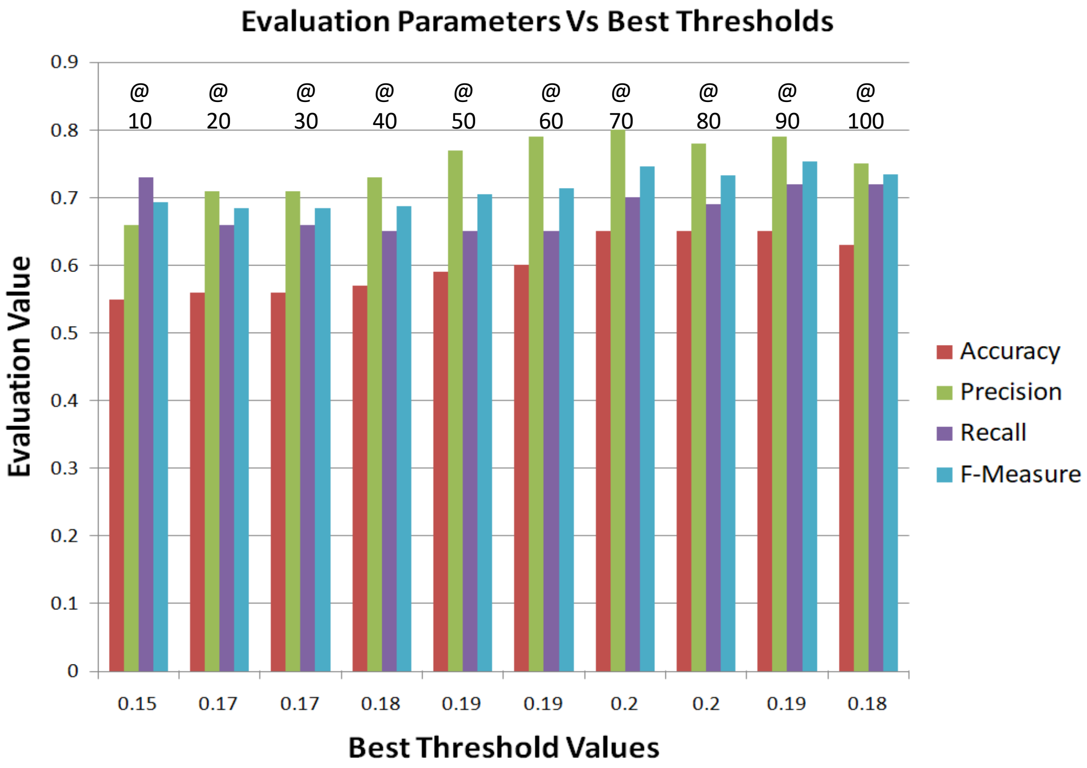


Figure 3.7: Evaluation Parameters Vs Best Threshold Values

Table 3.6: J.UCS Dataset Statistics

Categories	**Number of Papers**
A	41
B	77
C	172
D	585
E	62
F	445
G	125
H	760
I	372
J	105
K	236
L	33
M	26
Total Number of Research Papers	1,460
Total Number of Categories	13
Total Number of Papers Containing Categories	1147
Total Number of Papers without Keywords' Section	31

3.3.3 Evaluation of Threshold on J.UCS Dataset

We have critically analyzed the results of second type of experiments on almost 12 times larger Journal of Universal Computer Science (J.UCS) dataset as compared to the first type of experiments on 100 research papers. It contains 1,460 research publications from the diversified domains of Computer Science. The statistics for J.UCS dataset are presented in Table 3.6. Number of research papers in each category is also provided in Table 3.6.This dataset contains 3,039 total numbers of papers' categories pairs and then these pairs are merged with respect to their categories.

After applying the pre-processing steps on articles' metadata parameters as explained in the section 3.2.1.The papers-wise terms frequency weights are computed by utilizing the algorithm presented in the Figure 3.4. Similarly, from the metadata of the articles, the category-wise terms frequency weights are computed by using the Category-wise Metadata Merger (CMM).). The Category-wise distinct keywords (terms) count is shown in the Figure 3.8.

We have computed the evaluation parameters results at threshold value 0.2 (best threshold value for first type of experiments) for different metadata such as *Title*,

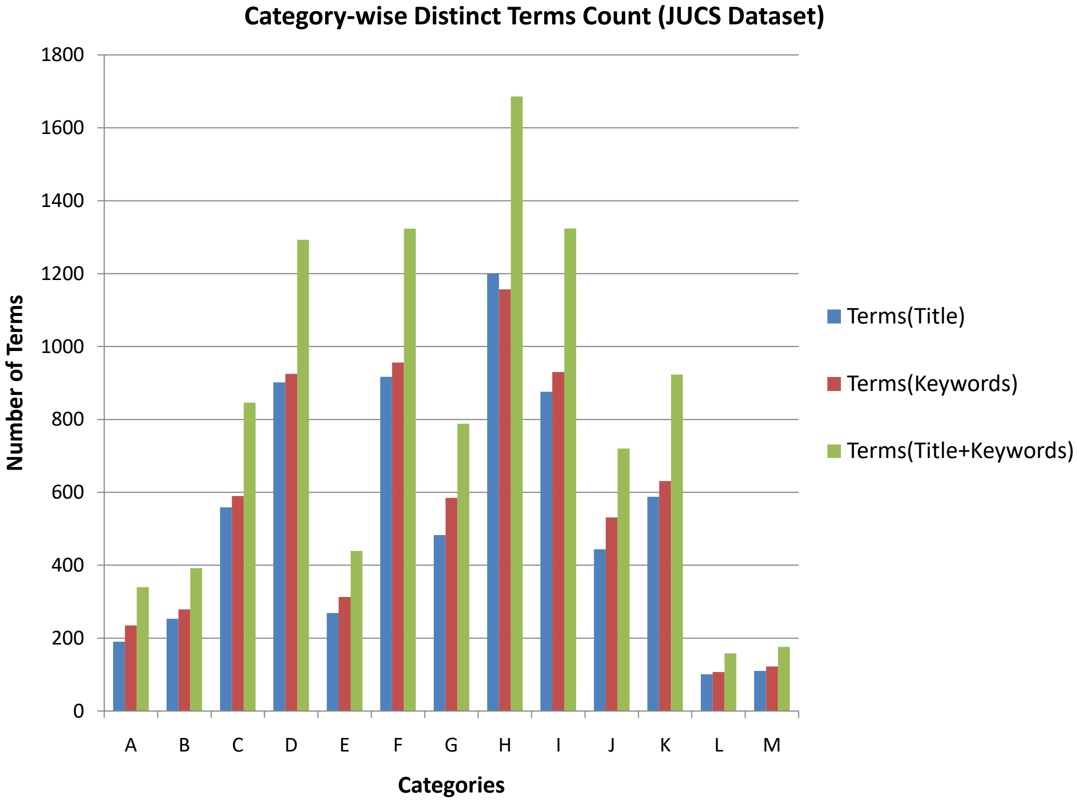
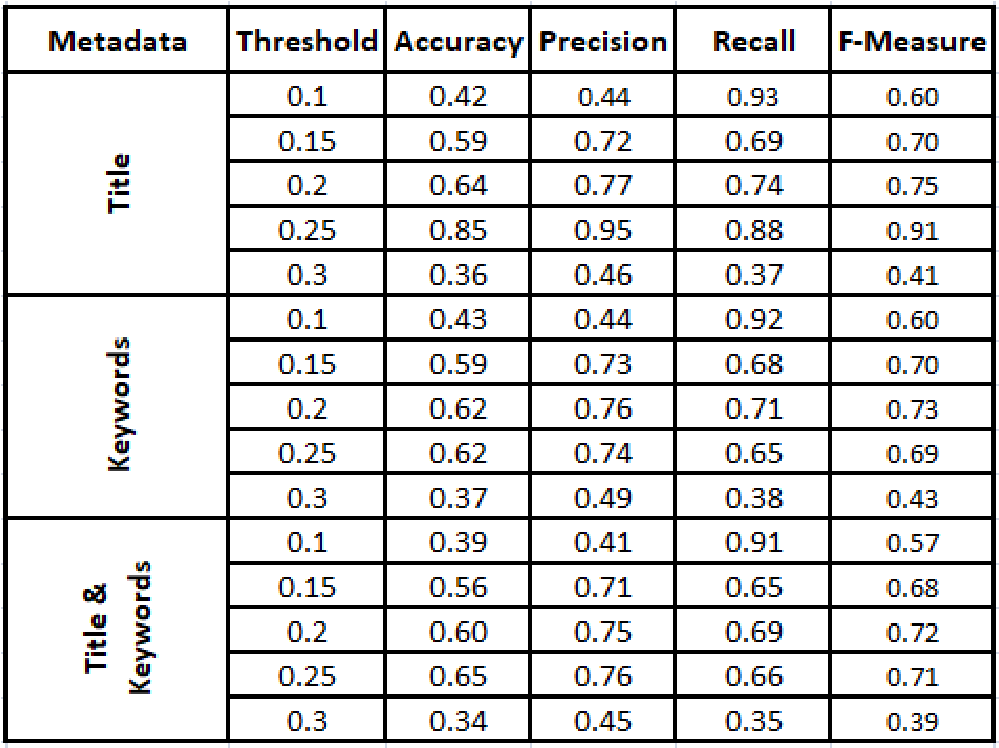


FIGURE 3.8: Category-wise Distinct Terms Count

Keywords and *Title* & *Keywords*. At benchmark threshold value 0.2, it is analyzed that for metadata (*Title*), the obtained F-measure value is 0.75, for metadata (*Keywords*) the achieved the F-measure value is 0.73 and for metadata (*Title* & *Keywords*) the achieved the F-measure value is 0.72. The results at this bigger dataset are almost quite close to the best threshold value (0.2) for the first type of experiments that is F-measure value is 0.75. Hence, experimental results show that the results are almost closed for both 100 research papers and for 12 time large dataset.

We have further investigated and evaluated experiments on different threshold values (0.1, 0.15, 0.2, 0.25, and 0.3) to analyze that at which threshold value MLC algorithm have produced best results and how far this threshold value could be more optimal than the benchmark of value 0.2. For the evaluation of experiments on these five threshold values (0.1, 0.15, 0.2, 0.25 and 0.3), the experiments are performed on J.UCS dataset in three ways such as: *Title*, *Keywords* and *Title* & *Keywords* of the research papers to find out the performance of our results and to make sure which metadata and threshold value yields better evaluation parameters results for multi-label document classification. J.UCS dataset results

Table 3.7: J.UCS Dataset Results



Metadata	Threshold	Accuracy	Precision	Recall	F-Measure
Title	0.1	0.42	0.44	0.93	0.60
	0.15	0.59	0.72	0.69	0.70
	0.2	0.64	0.77	0.74	0.75
	0.25	0.85	0.95	0.88	0.91
	0.3	0.36	0.46	0.37	0.41
Keywords	0.1	0.43	0.44	0.92	0.60
	0.15	0.59	0.73	0.68	0.70
	0.2	0.62	0.76	0.71	0.73
	0.25	0.62	0.74	0.65	0.69
	0.3	0.37	0.49	0.38	0.43
Title & Keywords	0.1	0.39	0.41	0.91	0.57
	0.15	0.56	0.71	0.65	0.68
	0.2	0.60	0.75	0.69	0.72
	0.25	0.65	0.76	0.66	0.71
	0.3	0.34	0.45	0.35	0.39

are presented in Table 3.7 below. The detailed description and analysis of above mentioned experiments and their results are as follows:

- Experiments on research papers' Titles
- Experiments on research papers' Keywords
- Experiments on both research papers' Titles and Keywords

3.3.3.1 Experiments on Research Papers' Titles

In Table 3.2, it is already described that in J.UCS dataset, there are 1,460 research papers and from these research papers, there are only 1,147 research articles have author's provided categories. Therefore, 1,147 papers' category pairs are generated in the pre-processing step (section 3.2.1).From these papers' category pairs; we have selected all distinct research papers and their *Titles*. By using this metadata (*Research Papers' Titles*), the Terms Frequency (TF) weights are computed for each research paper by using the algorithm presented in the Figure 3.4 and TF weights for each category are computed by using the algorithm presented in the Figure 3.3. After the completion of pre-processing step, these paper-wise TF weights and category-wise TF weights are given as an input to the Multi-Label Classifier (MLC) algorithm (see Figure 3.5) for the prediction of research papers categories. The MLC algorithm predicts a set of categories on the basis of different threshold values. It has been described above that the experiments are performed on five different threshold values and result is presented in Table 3.7. For the performance evaluation of these predicted results, the evaluation parameters are utilized as described above in section 3.3.1. In the Figure 3.9), J.UCS dataset results are presented according to the evaluation parameters for the multi-label document classification. The evaluation parameter values are presented on Y-axis and threshold values on X-axis. Based on in-depth analysis of the results of J.UCS dataset for metadata (*Title*), our findings are listed below:

- At a threshold value of 0.25, the performance of MLC algorithm is best among all other threshold values, for all evaluation parameters such as: accuracy, precision and F-measure except for the recall parameter which performed best on threshold value 0.1.

- The performance of the MLC algorithms is decreased as the threshold value is decreased from 0.25 to 0.1. This decrease in evaluation parameter values is reported due to the large number of predicted categories as compared to the original categories of the research papers.

- Similarly, when the threshold value is increased from 0.25 to higher, the performance of MLC algorithm became decrease due to very low numbers of predicted categories.

3.3.3.2 Experiments on Research Papers' Keywords

In this section, the experiments by using metadata (*Research Papers' Keywords*) from the research papers of the J.UCS dataset are presented. The *Keywords* are extracted from the authors provided keyword section in the research papers and assigned research papers' categories to these keywords. Same process has been followed as discussed above metadata (*Research Papers' Title*) for pre-processing task and for prediction of the categories for the research papers by using the MLC algorithm. For the performance evaluation of these predicted results, the same standard evaluation parameters (i.e., accuracy, precision, recall, and F-measure)

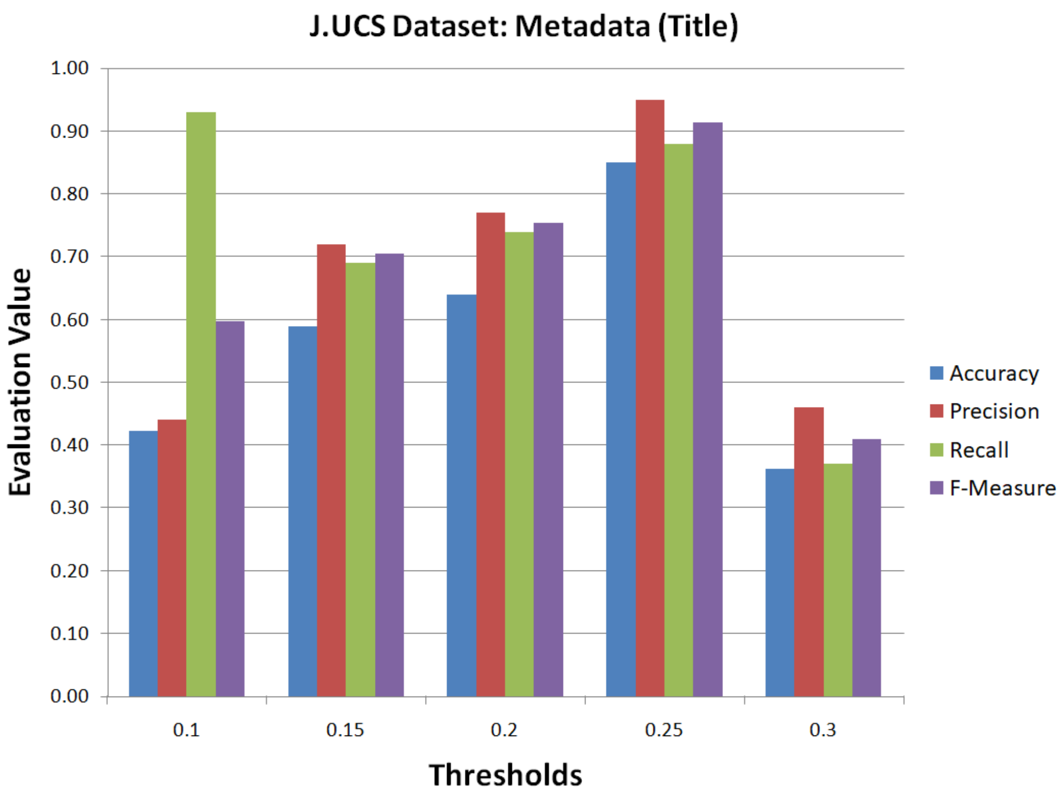


FIGURE 3.9: J.UCS Dataset: Evaluation Parameters Results for Metadata (*Title*)

are utilized. In the Figure 3.10, J.UCS dataset results for metadata (*Keywords*) are presented and evaluation parameters values are plotted on Y-axis and threshold values on X-axis. After the in-depth analysis of the results of J.UCS dataset for metadata (*Keywords*), the following points are observed:

- At threshold value 0.2, the performance of MLC algorithm is the best among all other threshold values for all evaluation parameters such as: accuracy, precision and F-measure except from the recall parameter which performed best for threshold value 0.1. This threshold value is much closer to the threshold value of 0.25 for metadata (*Title*) which produced best results at this threshold value and is close to that threshold value of 0.17 for the first experiments for 50 research papers.

- The performance of the MLC algorithms became decrease as the threshold value is decreased from 0.2 to 0.1. This performance decreases due to large number of predicted categories as compared to the original categories of the research papers.

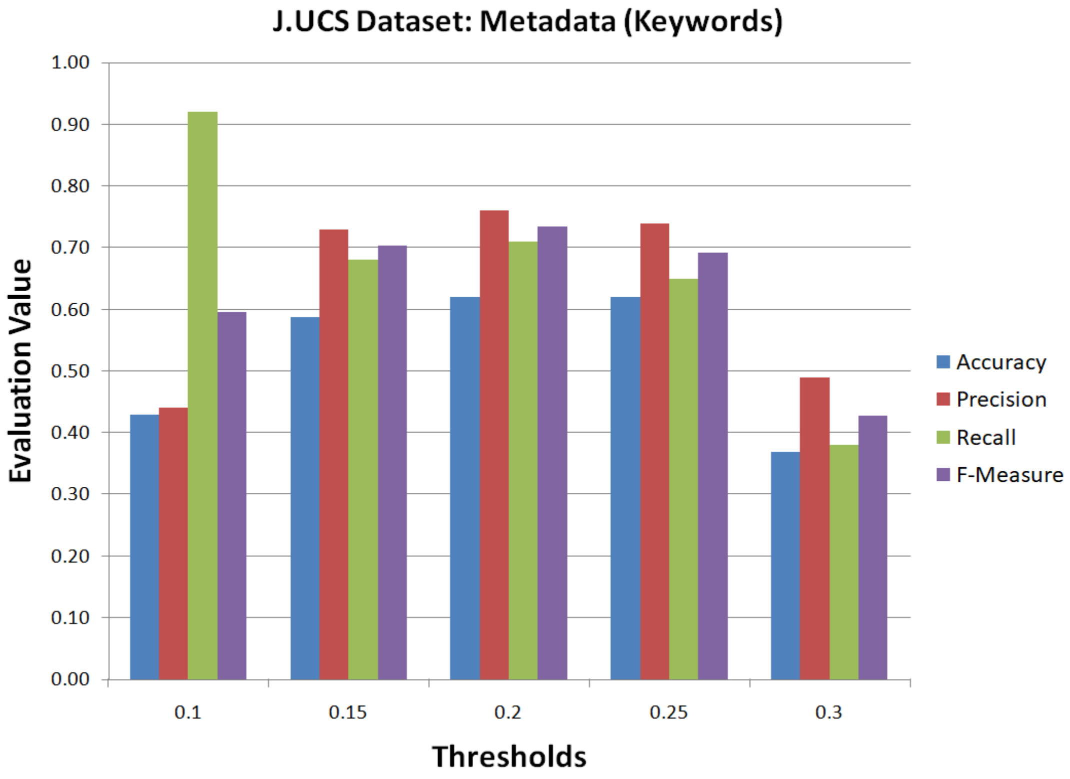


FIGURE 3.10: J.UCS Dataset: Evaluation Parameters Results for Metadata (*Keywords*)

- Similarly, when the threshold value is increased from 0.2 to higher, the performance of MLC algorithm became decrease due to small number of predicted categories.

- We have critically analyzed that why *Keywords* results are relatively low as compared to the *Title*? In our experiments, total numbers of research papers are 1,147 from which all contain the *Title* parameter. But in case of *Keywords*, there are 31 research papers which do not contain author provided keywords. Consequently, it has affected the performance of the results predicted by using the *Keywords* of the research papers only. Furthermore, the author provided *Keywords* are very generic as compared to the words represented in the *Titles* of research papers.

3.3.3.3 Experiments on Research Papers' Titles & Keywords

In the above two sections, the evaluation of the performance *Title* and metadata *Keywords* is presented. In this section, the experiments performed on collective metadata parameters *Title* & *Keywords* from the research papers of the J.UCS dataset are presented. All the *Title* & *Keywords* from title and authors provided keyword section are extracted. Same process has been followed as discussed above for metadata (*Research Papers' Title*) for pre-processing step and for the prediction of the categories for the research papers by using the MLC algorithm. For the performance evaluation of these predicted results, in the Figure 3.11, J.UCS dataset results for metadata (*Title* & *Keywords*) are presented and the evaluation parameter values are plotted on Y-axis and threshold values on X-axis. The in-depth analysis of the results for J.UCS dataset for *Title* & *Keywords* yielded following points:

- At threshold value 0.25, the performance of MLC algorithm is the best among all other threshold values for all evaluation parameters such as: accuracy, precision and F-measure except for the recall parameter which performed best on threshold value 0.1. This threshold value is much closer to the threshold value of 0.2 for *Keywords* which produced best results at this threshold value. But this threshold value is exactly equal to the threshold value for *Title*. Consequently, at threshold value 0.25, the performance of the MLC algorithm is best for the evaluation parameters.

- For *Title* & *Keywords* the performance of the MLC algorithms became decrease when the threshold value is decreased from 0.25 to 0.1. This is due to large number of predicted categories as compared to the original categories of the research papers. Similar behavior is also reported for metadata *Title*.

- Similarly, when the threshold value is increased from 0.25 to higher, the performance of MLC algorithm became decrease. This decrease in evaluation parameters values is reported due to very low numbers of predicted categories.

- We have critically analyzed that why metadata (*Title* & *Keywords*) results are relatively low as compared to the metadata (*Title*)? When we added metadata (*Keywords*) with the metadata (*Title*) then the metadata (*Title* & *Keywords*) becomes somehow noisy and due to this noisy metadata (*Title* & *Keywords*), the TF weights for these terms are increased. Due to this reason, at low threshold values, MLC algorithm predicts more categories as compared to the predicted categories of metadata (*Title*).

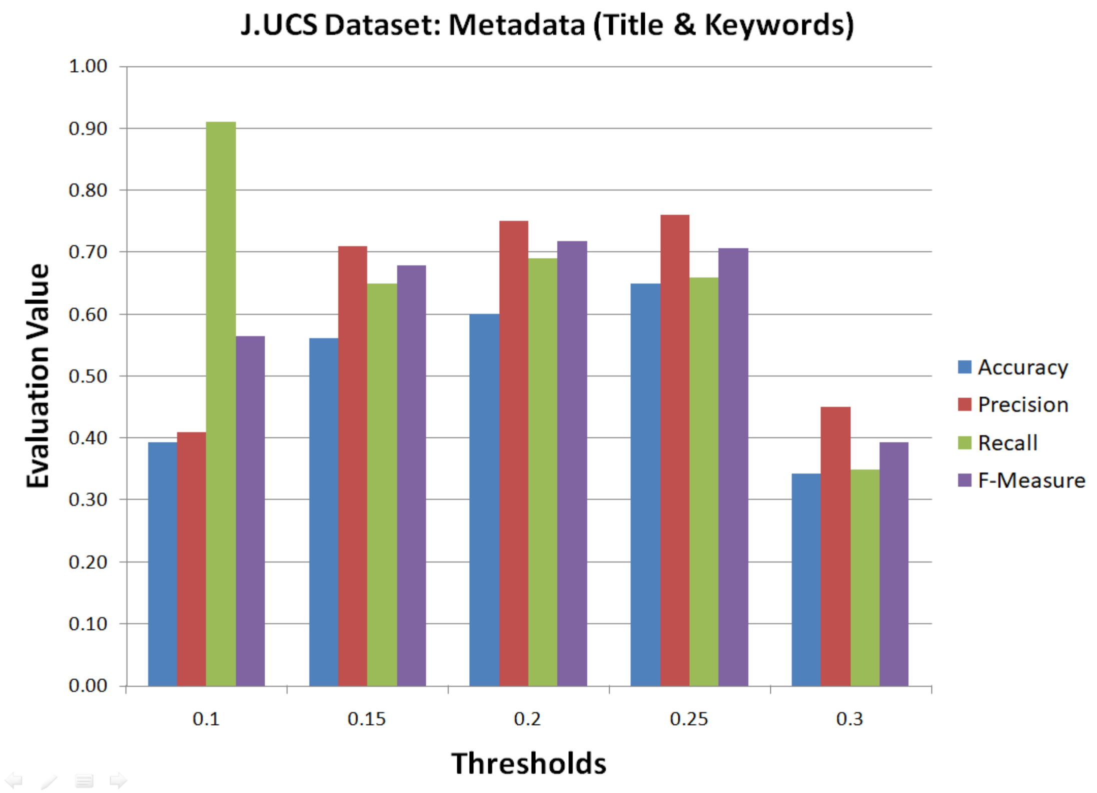


Figure 3.11: J.UCS Dataset: Evaluation Parameters Results for Metadata (*Title & Keywords*)

The experiments are performed on the J.UCS dataset and performance evaluation of these experimental results is conducted by using the evaluation parameters proposed by [62] for the multi-label document classification. The experimental results are analyzed in detail by applying different threshold values to MLC algorithm. It yields different results at different threshold values.

- MLC algorithm performance is significantly good at threshold value 0.25 as shown in the above mentioned experimental results figures for the J.UCS dataset. At this threshold value, MLC algorithm's performance is good for both metadata (*Title*) and metadata (*Title & Keywords*).
- The results of MLC algorithm are good in the range from 0.2 to 0.25 threshold values. When we choose threshold value less than 0.2 it predicts more categories which may decrease the accuracy as well as precision values. On the other hand, it may increase the recall values. If the threshold value has increased from 0.25 then the number of predicted categories is too low which results in decreasing all evaluation parameter values.

Hence for this particular dataset, the most optimum threshold value is 0.25 which produces most significant results for multi-label document classification. For benchmark threshold value of 0.2, the metadata (*Title*) achieved F-measure value of 0.75, however, in this dataset; the best F-measure achieved was 0.91. Therefore, we are 0.16 points away using the fine tuned threshold value. The threshold value of 0.2 will further be evaluated on ACM dataset. Then at the end we will be able to conclude the best threshold value. Similarly, MLC algorithm yields better results at threshold value 0.25 for the metadata (*Keywords*) with F-measure of 0.69 with the error rate of 0.04 for threshold value 0.2. MLC algorithm also yields better results at threshold value 0.25 for the metadata (*Title* & *Keywords*) with F-Measure value of 0.71 with the error rate of 0.01 for threshold value 0.2. Hence, the error rate with respect to the benchmark optimum threshold value 0.2 is very small and may be negligible. We can say that at threshold value 0.2, the proposed MLC algorithm gives better results for metadata (*Title*) in terms of multi-label document classification. We have further validated the optimal threshold value (0.2) on 15 time bigger dataset (ACM dataset) as compared to the first experiment on 100 papers. The detailed description of ACM dataset and the evaluation of optimum threshold value are presented in the following section.

3.3.4 Evaluation of Threshold on ACM Dataset

The Association for Computing Machinery (ACM) dataset [18], contains 86,116 research publications from different workshops, conferences and journals. It encompasses 11 distinct categories at the top level of ACM computing classification system (CSS). Statistics for ACM dataset is presented in Table 3.8. This dataset contains research papers which belong to one category or multiple categories as presented in Table 3.2. Number of research papers in each category is also provided in Table 3.8. This dataset contains 137,679 total numbers of papers' category pairs and then these pairs are merged with respect to their categories. After applying the pre-processing steps on the metadata of the research papers from ACM dataset, the papers-wise terms frequency weights are computed by using the algorithm presented in the Figure 3.4. Similarly, from the metadata of the research papers, the category-wise terms frequency weights are computed by using the Category-wise Metadata Merger (CMM) algorithm which is explained above in section 3.2.2 (Figure 3.3). Category-wise distinct keywords (terms) count is shown in the Figure 3.12. For the evaluation of ACM dataset's experiments (15

Table 3.8: ACM Dataset Statistics

Categories	**Number of Papers**
A	648
B	9,904
C	12,314
D	31,301
E	566
F	8,243
G	4,217
H	30,778
I	24,458
J	1,377
K	13,873
Total Number of Research Papers	86,116
Total Number of Categories	11
Total Number of Papers Containing Categories	54,994
Total Number of Paper's Category Pairs	1,37,679
Total Number of Different Workshops, Conferences and Journals	2,240

time bigger dataset than first experiments for 100 research papers), similar threshold values (as for J.UCS dataset) 0.1, 0.15, 0.2, 0.25 and 0.3 have been evaluated for the ACM dataset. This dataset contains 86,116 research publications from 2,240 different workshops, conferences and journals. From these 86,116 research publications there are only 54,994 research publications which entail authors provided categories or classes and from these 54, 994 research papers we have generated 137,679 papers' category pairs. These statistics for ACM dataset are already presented above in Table 3.8.

We have critically analyzed the results of third type of experiments on 15 time larger dataset as compared to the first type of experiments on 100 research papers. The evaluation parameters results at benchmark threshold of value 0.2 are calculated for different metadata parameters such as *Title*, *Keywords* and *Title & keywords.* At this threshold value of 0.2, the F-measure value of 0.92 is achieved for metadata (*Title*), F-measure value of 0.73 for metadata (*Keywords*) and F-measure value of 0.86 for metadata (*Title & keywords*) for the proposed MLC algorithm. The experimental results are further evaluated and compared on different threshold values (0.1, 0.15, 0.2, 0.25, and 0.3) to validate that at which threshold value MLC algorithm produces best results for this larger dataset. For the evaluation of experiments on these five threshold values (0.1, 0.15, 0.2, 0.25 and 0.3), the experiments are performed on ACM dataset. The results of these experiments for ACM dataset are presented in Table 3.9 below. These experimental results are presented in three ways to analyze the effectiveness of our results and

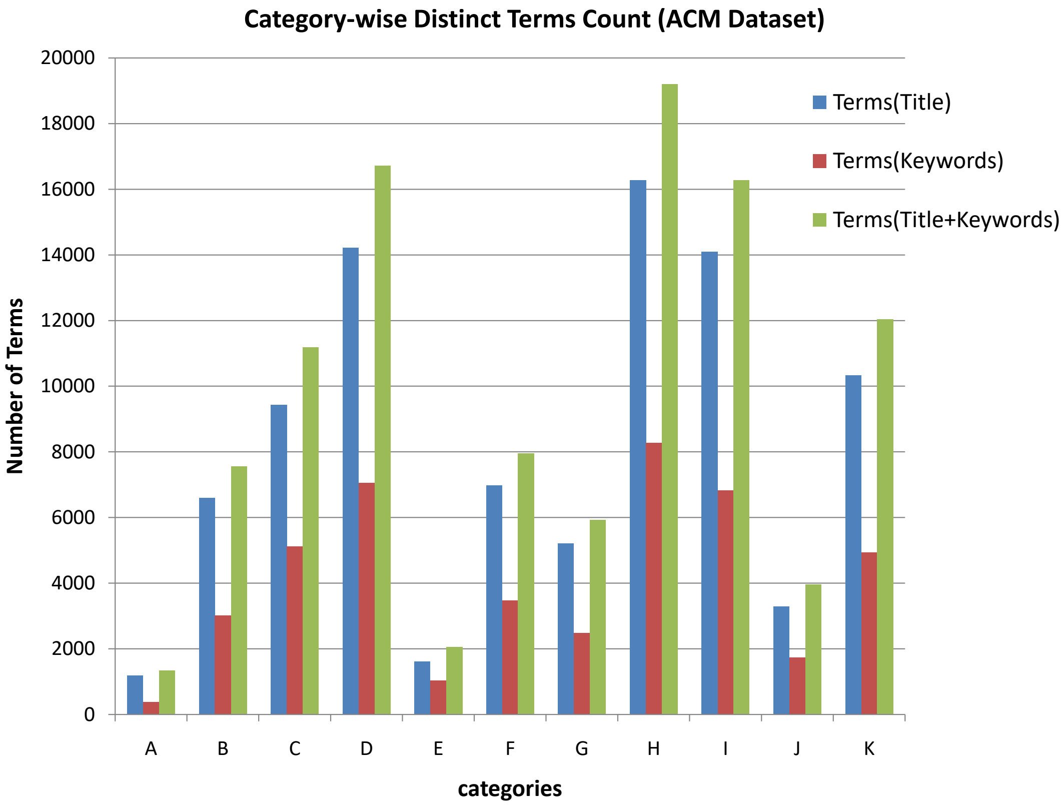


Figure 3.12: Category-wise Distinct Terms Count (ACM Dataset)

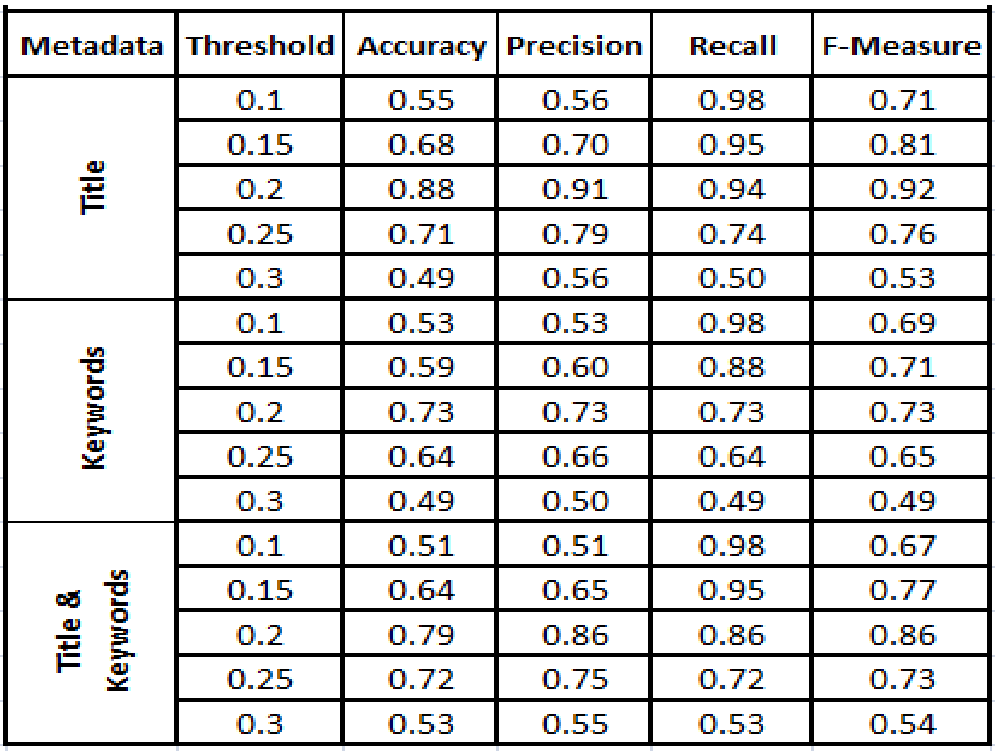
to analyze that which metadata and threshold value yields better evaluation parameters results for multi-label document classification. The detailed description and analysis of these experiments and their results are as given below:

- Experiments on research papers' Titles
- Experiments on research papers' Keywords
- Experiments on both research papers' Titles and Keywords

3.3.4.1 Experiments on Research Papers' Titles

In Table 3.8, it is already described that in ACM dataset, there are 86,116 research papers and from these research papers, only 54,994 research papers have author's provided categories. Therefore, 137,679 papers' category pairs are generated in the pre-processing step (section 3.2.1). From these papers' category pairs, first 1500 research papers and their Titles are selected for evaluation of the results. By using this metadata (*Research Papers' Titles*), the Terms Frequency (TF) weights

Table 3.9: ACM Dataset Results



Metadata	Threshold	Accuracy	Precision	Recall	F-Measure
Title	0.1	0.55	0.56	0.98	0.71
	0.15	0.68	0.70	0.95	0.81
	0.2	0.88	0.91	0.94	0.92
	0.25	0.71	0.79	0.74	0.76
	0.3	0.49	0.56	0.50	0.53
Keywords	0.1	0.53	0.53	0.98	0.69
	0.15	0.59	0.60	0.88	0.71
	0.2	0.73	0.73	0.73	0.73
	0.25	0.64	0.66	0.64	0.65
	0.3	0.49	0.50	0.49	0.49
Title & Keywords	0.1	0.51	0.51	0.98	0.67
	0.15	0.64	0.65	0.95	0.77
	0.2	0.79	0.86	0.86	0.86
	0.25	0.72	0.75	0.72	0.73
	0.3	0.53	0.55	0.53	0.54

are computed for each research paper by using the algorithm presented in the Figure 3.4.The TF weights for each category are also computed by using the algorithm presented in the Figure 3.3. For the performance evaluation of predicted results for metadata (Title), the standard evaluation parameters as described above in section 3.3.1 are utilized. In the Figure 3.13, ACM dataset results are presented according to the evaluation parameters for the multi-label document classification. The evaluation parameter values are plotted on Y-axis and threshold values on X-axis. After the critical analysis of the results of ACM dataset for metadata (*Title*), we have concluded that:

- At threshold value 0.2, the performance of MLC algorithm is the best among all other threshold values for all evaluation parameters such as: accuracy, precision and F-measure except for the recall parameter which performed best on threshold value 0.1.

- The performance of the MLC algorithms gradually became decrease as the threshold value decreased from 0.2 to 0.1. This decrease in evaluation parameters values is reported due to large number of predicted categories as compared to the original categories of the research papers.

- Similarly, when the threshold value is increased from 0.2 to higher, the evaluation parameters values gradually decrease due to small number of predicted categories.

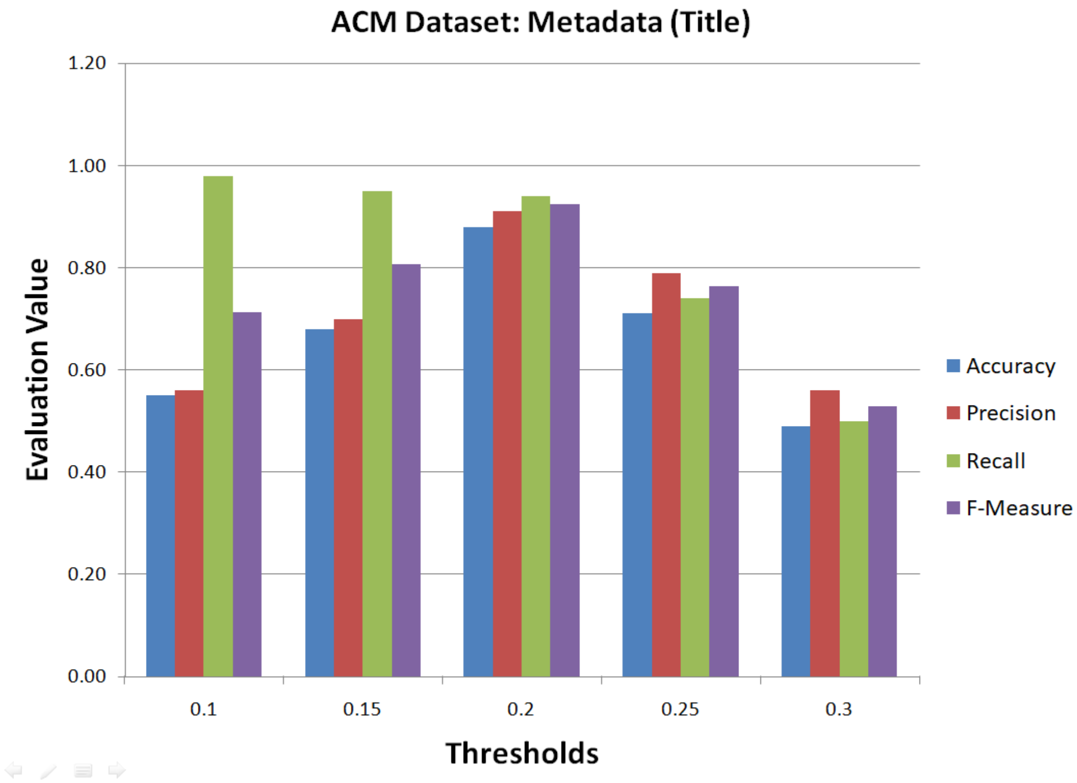


Figure 3.13: ACM Dataset: Evaluation Parameters Results for Metadata (*Title*)

3.3.4.2 Experiments on Research Papers' Keywords

In this section, the experiments performed by using the metadata (*Research Papers' Keywords*) from the research papers of the ACM dataset are presented. All keywords are extracted from the authors provided keywords' section in the research papers and assigned categories of the research papers' to these keywords. Same process has been followed as discussed above for metadata (*Research Papers' Title*) for preprocessing step and for the prediction of the categories for the research papers by using the MLC algorithm. To evaluate the performance of these predicted results the standard evaluation parameters (i.e., accuracy, precision, recall and F-measure) are utilized. In the Figure 3.14, ACM dataset results for metadata (*Keywords*) are presented and the evaluation parameter values are

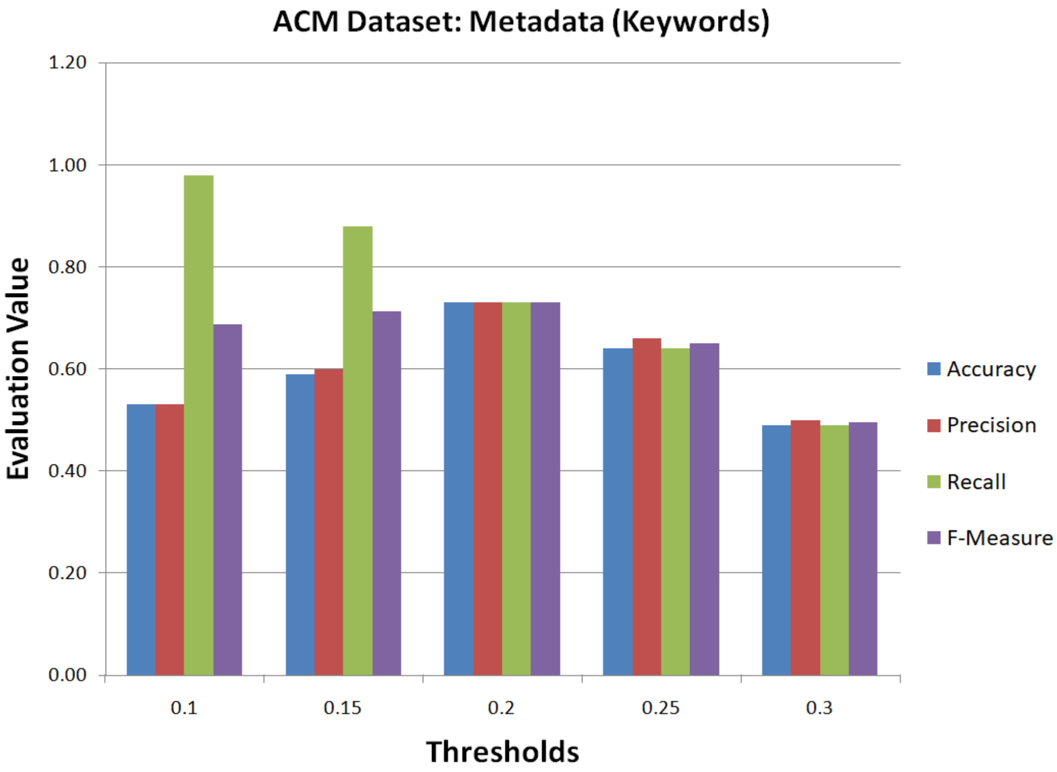


Figure 3.14: ACM Dataset: Evaluation Parameters Results for Metadata (*Keywords*)

plotted on Y-axis and threshold values on X-axis. Based on the in-depth analysis of the results of J.UCS dataset for metadata (Keywords), we have concluded that:

- At threshold value 0.2, the performance of MLC algorithm is the best among all other threshold values for all evaluation parameters such as: accuracy, precision and F-measure except for the recall parameter which performed best on threshold value 0.1same as metadata (*Title*) performance.
- The performance of the MLC algorithms gradually became decrease as the threshold value decreases from 0.2 to 0.1. This decrease in evaluation parameters valuse is due to large number of predicted categories as compared to the original categories of the research papers.
- Similarly, when the threshold value is increased from 0.2 to higher, the evaluation parameters values gradually decreased due to small number of predicted categories.
- We have critically analyzed that why metadata (*Keywords*) results are relatively low as compared to the metadata (Title)? In experiments, total

numbers of research papers are 1,500 and all of them contribute for the prediction of the categories for the metadata (Title) results as the *Title* of every research article is available. However, in case of metadata (*Keywords*), there are 90 research papers which have not been assigned keywords by the authors. Therefore, the proposed approach's performance is low for metadata (*Keywords*) as compared to the performance of the metadata (*Title*).

3.3.4.3 Experiments on Research Papers' Titles and Keywords

In the above two sections, the performance evaluation of the metadata (*Title*) and metadata (*Keywords*) is presented. In this section, the experiments on collectively metadata parameters (*Title* & *Keywords*) from the research papers of the ACM dataset are presented. All the *Title* & *Keywords* are extracted from title and authors provided keyword section in the research papers. The same process has been followed as discussed above for metadata (*Research Papers' Title*) for preprocessing task and for the prediction of the categories for the research papers by using the MLC algorithm. For the performance evaluation of these predicted results, in the Figure 3.15, ACM dataset results for metadata (*Title* & *Keywords*) are presented and evaluation parameters values are plotted on Y-axis and threshold values on X-axis. Based on the in-depth analysis of the results of ACM dataset for metadata (*Title* & *Keywords*), we have concluded that:

- At threshold value 0.2, the performance of MLC algorithm is the best among all other threshold values for all evaluation parameters such as: accuracy, precision and F-measure except for the recall parameter which performed best on threshold value 0.1 same as metadata (*Title*) and metadata (*Keywords*). Consequently, threshold value 0.2 is an optimum value in which the performance of MLC algorithm is the best for the evaluation parameters.
- For metadata (*Title* & *Keywords*) the performance of the MLC algorithms gradually became lower as the threshold value decreased from 0.2 to 0.1. The decrease in evaluation parameters values is reported due to large number of predicted categories as compared to the original categories of the research papers. Similar behavior is reported for both metadata (*Title*) and metadata (*Keywords*).

- Similarly, when the threshold value is increased from 0.2 to higher, the performance of MLC algorithm became low due to small number of predicted categories as we have examined for both metadata (*Title*) and metadata (*Keywords*).

- We have critically analyzed that why metadata (*Title* & *Keywords*) results are relatively low as compared to the metadata (*Title*) and metadata (*Keywords*)? When we added metadata (Keywords) with the metadata (Title) then metadata (*Title* & *Keywords*) becomes somehow noisy and due to this noisy metadata (*Title* & *Keywords*) the TF weights for these terms get increased. Due to this reason, at low threshold values, MLC algorithm predicts more set of categories as compared to the predicted set of categories of metadata (*Title*) and metadata (*Keywords*). Similarly, when we increase the threshold values for the metadata (*Title* & *Keywords*) then there is a very low ratio of correct predicted set of categories because at high threshold value, MLC predicts small number of categories.

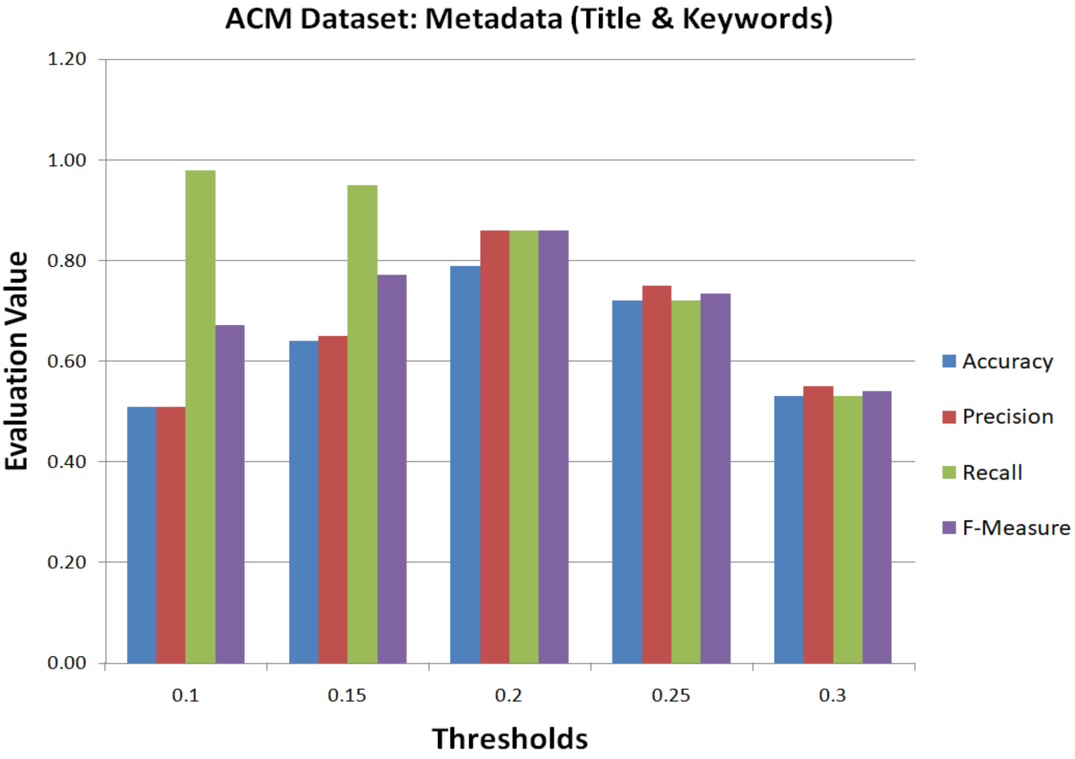


FIGURE 3.15: ACM Dataset: Evaluation Parameters Results for Metadata (*Title* & *Keywords*)

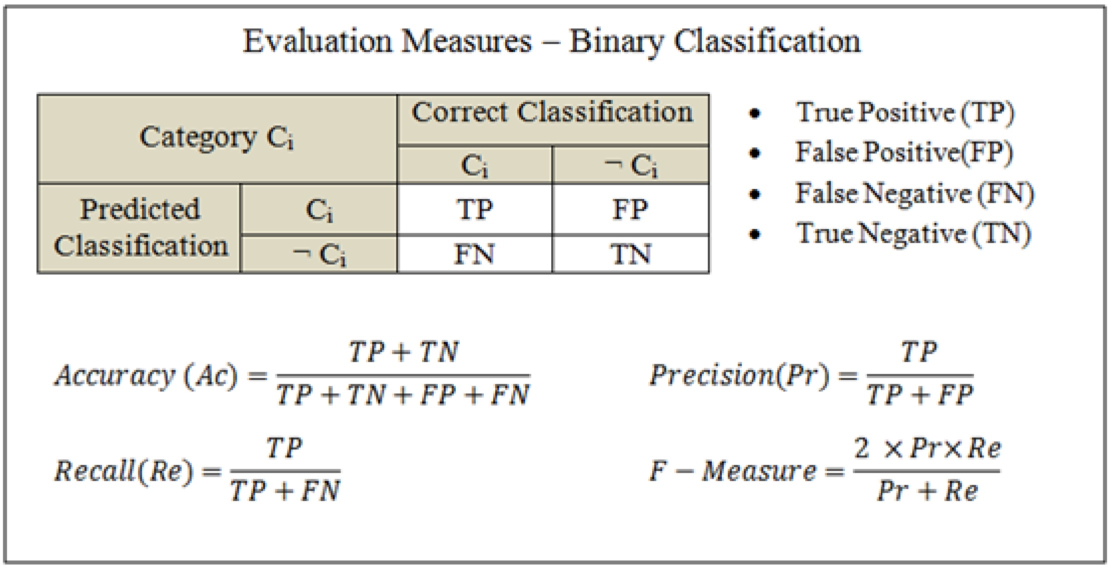
The experiments are performed on the ACM dataset for evaluation of these experimental results is conducted by using the evaluation parameters proposed by [62] for the multi-label document classification. We have critically analyzed these experimental results by applying different threshold values to our MLC algorithm. It yields different results at different threshold values.

- MLC algorithm performance is significantly good at threshold value 0.2 as shown in the above mentioned experimental results figures for the ACM dataset. At this threshold value, MLC algorithm's performance is good for all metadata (*Title*), metadata (*Keywords*) and metadata (*Title & Keywords*).
- The results of MLC algorithm are good when the threshold value ranges from 0.2 to 0.25. If we pick a threshold value less than 0.2, it predicts more categories which may decrease in accuracy as well as precision values. On the other hand it may increases the recall value and if we increase threshold value from 0.25 then the number of predicted categories are very small which result in decreasing all evaluation parameters values.

It has been noticed from above experiments that the proposed MLC algorithm produced better results at threshold value 0.2 for all metadata (*Title*), metadata (*Keywords*) and metadata (*Title & Keywords*).
Hence, three different types of experiments are performed one on small dataset (100 research papers) and two experiments on large datasets, one is 12 times (J.UCS dataset) and other is 15 times (ACM) bigger than the dataset on which threshold was tuned. Above experimental results show that for first and third types of experiment the most optimum threshold value is same that is 0.2. But for the second type of experiments, the results at threshold value 0.25 remained the best; however, the results are close enough to threshold value 0.2. Therefore, by doing comprehensive experiments on different data sizes, it has been identified that at threshold value 0.2; the proposed model yields significant results for diversified datasets by exploiting only metadata of the research papers. Similarly, proposed model yields significant numbers of results for metadata (*Title*) at threshold value 0.2 as compared to the other metadata such as metadata (*Keywords*) and metadata (*Title & Keywords*).

3.4 Single-Label Document Classification

The proposed approach is evaluated on both datasets for single-label document classification as well. The proposed MLC algorithm predicts only the category which has highest membership value among all other categories. If there is a tie among more than one category with their highest membership value then all those categories are considered. Unlike multi-label classification, the threshold value is not required for single-label document classification. It's a binary decision whether a research paper is correctly classified or not. For evaluation, a confusion matrix for binary decisions is utilized which is shown in the Figure 3.16. The experimental results have been evaluated for 1,147 research papers from the J.UCS dataset. We have assumed that the categories annotated by the authors to these research papers are correct and we have conducted our experiments on all 1,147 distinct research papers' category pairs. In the Figure 3.17, the metadata parameters on X-axis and evaluation parameters values form are plotted on Y-axis. We have critically analyzed the results of all evaluation parameters with respect to the different metadata values and pointed out the following observations:



Evaluation Measures – Binary Classification

Category C_i		Correct Classification	
		C_i	$\neg C_i$
Predicted Classification	C_i	TP	FP
	$\neg C_i$	FN	TN

- True Positive (TP)
- False Positive(FP)
- False Negative (FN)
- True Negative (TN)

$$Accuracy\ (Ac) = \frac{TP + TN}{TP + TN + FP + FN}$$

$$Precision(Pr) = \frac{TP}{TP + FP}$$

$$Recall(Re) = \frac{TP}{TP + FN}$$

$$F - Measure = \frac{2 \times Pr \times Re}{Pr + Re}$$

Figure 3.16: Binary Classification Evaluation Measures

- An MLC algorithm performs well against all the evaluation parameters on different metadata parameters and almost 0.87 times it correctly classifies the research papers.

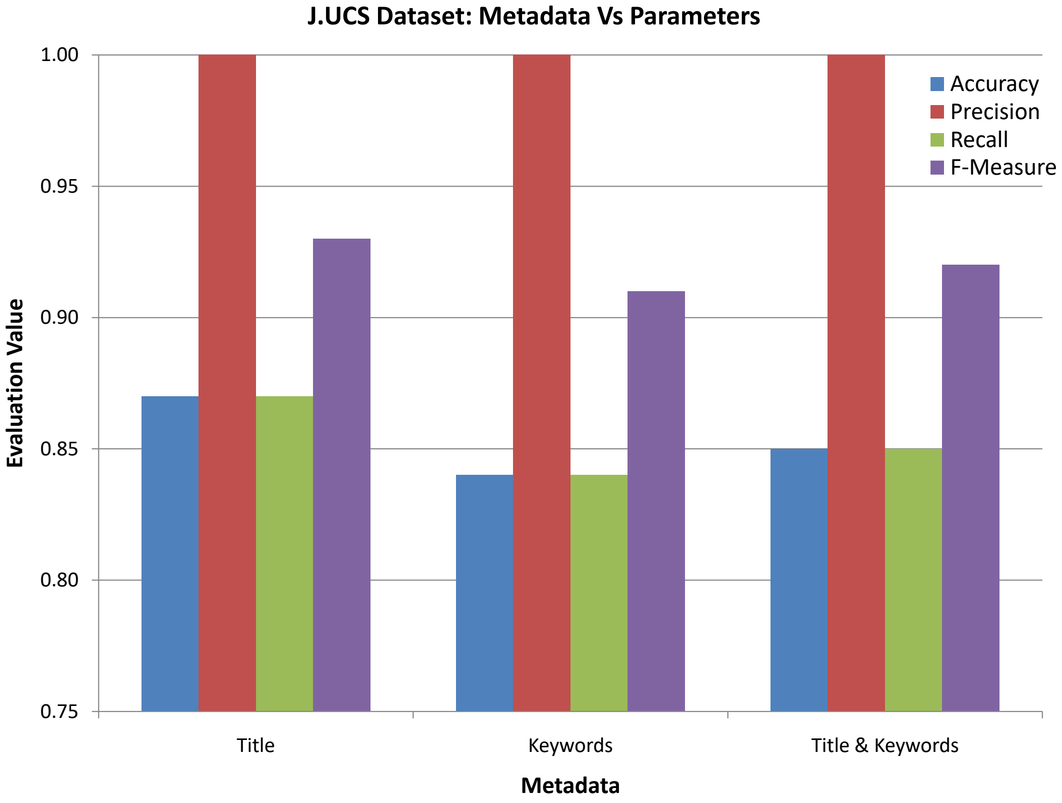


FIGURE 3.17: J.UCS Dataset: Metadata Vs Evaluation Parameters

- We have assumed that the annotated categories are correct that's why precision of our MLC algorithm is 100% as all predicted categories are correct and there is no False Positive (FP).
- Proposed approach's accuracy and recall have same values because there is no True Negative (TN) value in the confusion matrix.

Similarly, the experiments are performed on ACM dataset and for these experiments we have taken 1,500 research papers from ACM dataset. Similar assumption has also been taken for these experiments that already categories annotated by authors to these research papers are correct and we have conducted our experiments on all 1,500 distinct research papers' category pairs. In the Figure 3.18, we have plotted different metadata parameters on X-axis and evaluation parameters values are plotted on Y-axis. After the critical analysis of these experiments with different metadata parameters we have concluded that:

- An MLC algorithm performs well against all the evaluation parameters on different metadata parameters and almost 88% times it correctly classifies the research papers.
- We have assumed that the annotated categories are correct that's why precision of our MLC algorithm is 100% as all predicted categories are correct and there is no False Positive (FP).
- Proposed approach's accuracy and recall have same values because there is no True Negative (TN) value in the confusion matrix.

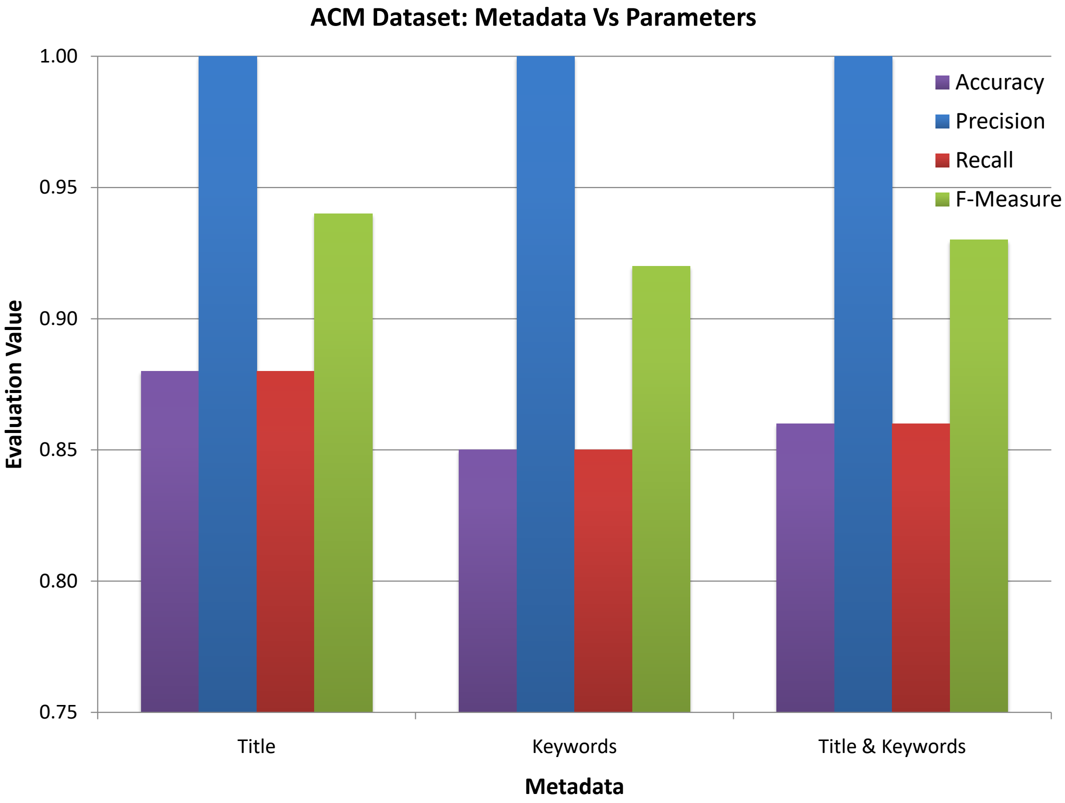


FIGURE 3.18: ACM Dataset: Metadata Vs Evaluation Parameters

3.5 Summary

This chapter presents a novel approach to classify research articles by applying comprehensive proposed mechanism. Two types of document classifications are performed (i) one is multi-label and other is (ii) single-label document classification. Different experiments are performed for multi-label as well as single-label document classification by exploiting only metadata like *Title*, *Keywords* and *Title & Keywords* of the research papers and evaluated on two diversified datasets ACM and J.UCS. For multi-label document classification, different experiments have been performed on different data sizes to find out the optimum threshold values and metadata for the proposed model to predict one or more than one category. The experimental results depicted that the proposed model achieves good results at the threshold value of 0.2 for 100 research papers as well as for 15 times larger dataset (ACM). The results of this threshold value are also very closer to the best results for 12 time larger dataset (J.UCS) with a very low error rate. Consequently, we have identified the optimum threshold value 0.2 and metadata (*Title*) at which proposed model yields the best results by exploiting only metadata of the research papers and perform multi-label document classification. These results for both single-label and multi-label document classification at threshold value 0.2, are compared with the state-of-the-art classification approaches and are presented in the Chapter 5.

Chapter 4

Multi-Label Document Classification based on Paper's References

The profusion of documents production at an exponential rate over the web has made it difficult for the scientific community to retrieve most relevant information against the query. The research community is busy in proposing innovative mechanisms to ensure the document retrieval in a flexible manner. The document classification is a core concept of information retrieval that classifies the documents into predefined categories. Though the idea of document classification was coined in 60's but has gained a paramount interest in the early 90's due to the increased availability of research articles in digital form [7]. The major portion of document classification techniques tackles the issue of research articles' classification.

These schemes perform either single-label (i.e., classifying the items into single class) classification or multi-label classification (i.e., classifying the items into more than one class) [21–23]. However, the state-of-the-art trend pertaining to single-label classification techniques is at large-scale in than multi-label classification. Since multi-labeling requires more extensive effort to produce the set of comprehensive features. We argue that a research article may belong to more than one category due to the linkage of diversified concepts with multiple domains, the argument is already validated via experiments on diversified datasets (see Chapter 3, Table 3.2). Hence, multi-label classification is a daunting but demanding at the same time. Most of the multi-label classification schemes yield low accuracy and classify research articles into a limited number of categories [14, 29, 30].

Can we perform multi-label classification to classify research articles into multiple classes using papers' references available openly on the Web? This chapter presents a novel framework based on multi-label classification, the insights of its implementation and evaluation to address the above question. As described earlier (see Chapter 3), this dissertation is focused at scrutinization of metadata potential due to its free availability on various platforms (i.e., IEEE , ACM , and Springer) than content based features. Now the important task is the selection of most suitable metadata parameters. The one logical and optimal parameter could be the citations of research papers. Diverse types of relationships exist between citing and cited documents [63]. The citation delineates a relationship between part and whole of the cited documents [24]. Moreover, the references are reckoned as quintessential indicator when it comes to recommending relevant research papers [13, 64, 65]. Therefore, we have selected the reference's section of research articles to map it into predefined categories. Different assumptions are presented below to signify the selection of reference parameter further.

1. Contemplate a scenario where you have an article that belongs to a specific topic, and you want to acquire more research articles on the same topic, one of the possible ways would be to look for references and citations of that particular paper because there is surety of relationship existence among them[24].

2. Most of the time, cited and citing work lie under the same category or topic. The articles cited by the authors usually belong to the similar categories.

3. The contemporary state-of-the-art approaches have not employed references of the papers to classify research documents.

The initial experimental phase of this framework is published in a reputed conference named as MEDES, an ACM Conference [10]. The proposed mechanism not only performs single-label classification, but also achieves multi-label classification. For the evaluations of proposed technique, two different datasets, (1) J.UCS [37] and (2) ACM [18] have been utilized. The statistics of both datasets are presented in Table 3.2.

4.1 Citation Based Category Identification (CBCI) Framework

The proposed framework is presented to classify research articles into multiple categories using reference's section of research papers is termed as "Citation Based Category Identification" (CBCI) [10]. This section presents the detailed overview of applied methodological steps. In the preprocessing, the Reference Extractor automatically extracts the references of the research papers from the web and made the Topic-Reference (TR) pairs and stored in the database as presented in the Figure 4.1. Similarly, the references of the test paper also extracted by the Metadata Extractor. The classifier then finds the similarity of each reference of the test paper with category wise all references in the dataset. After finding the similarity value for each category against all references of the test paper, we also computed the total similarity value for all categories. Then the classifier will predict the set of categories for the test paper based on some threshold value. The updater module will also update the dataset according to the predicted categories and test paper references. The graphical depiction of the proposed framework is presented in Figure 4.1. The sub-sections are organized as follows: Section 4.1.1, presents the mechanism to identify the overlapping categories. Section 4.1.2 illustrates the common and important steps of pre-processing to make the documents ready for input. Section 4.1.3 elaborates the CBCI algorithm and it's working.

4.1.1 Document Representation

The experiments on diversified dataset show that there are significant numbers of papers that belong to multiple categories as shown in Table 3.2. There is a great possibility that a research paper is partially associated with one category and is partially associated to other categories. We have developed a mechanism for the identification of such type of overlapping which is already described in Eq. 3.1 (Chapter 3).

4.1.2 Pre-processing

The data pre-processing step is obligatory to make the input file ready for experiments. First of all, the metadata from each research paper is extracted which is

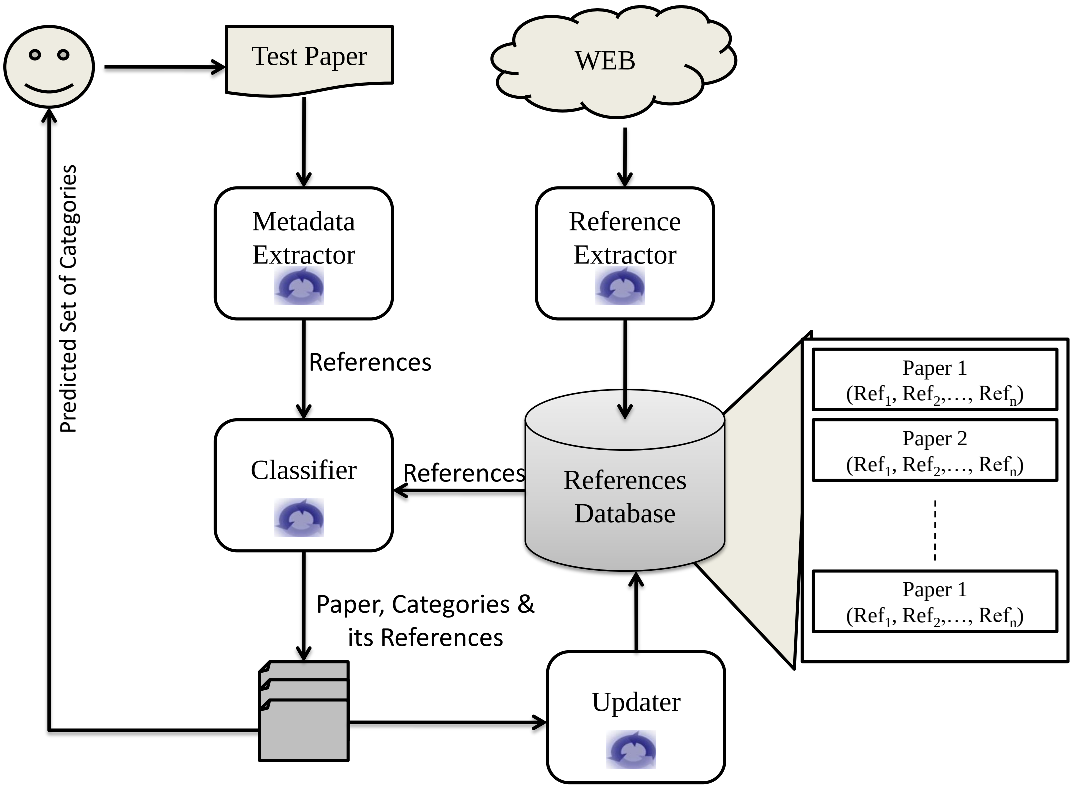

Test Paper
WEB
Metadata Extractor
Reference Extractor
References
Classifier
References
References Database
Paper 1 (Ref$_1$, Ref$_2$,…, Ref$_n$)
Paper 2 (Ref$_1$, Ref$_2$,…, Ref$_n$)
Paper 1 (Ref$_1$, Ref$_2$,…, Ref$_n$)
Paper, Categories & its References
Updater
Predicted Set of Categories

Figure 4.1: Citation Based Category Identification (CBCI) Framework

being used for the classification of research papers. The extracted metadata contains author, category (topic) and references as depicted in the Figure 4.2. Author information contains a list of all authors provided in the research paper. Category information contains the list of all categories provided by the authors of research papers. References contain all the extracted references from the article's reference section. On the basis of extracted metadata parameters, the Topic (category) and Reference (TR) pairs are generated and stored in the database to assign a topic (category) to the input article. In CBCI framework, references extractor extracts all the references from the web links provided by J.UCS database [37] and stored in the database according to their research papers. We have already extracted the research papers and their author annotated categories (topics). Most of the time, the cited and citing papers belong to the same category. For instance, if a research paper belongs to the topic "Information System" (Category H), then there is a high probability that all of its references would also be from the topic "Information System" (Category H). Relying on this assumption, the Topic-Reference (TR) pairs are generated from the references stored in the database along with the matching category (topic) of their research papers once for all the research papers

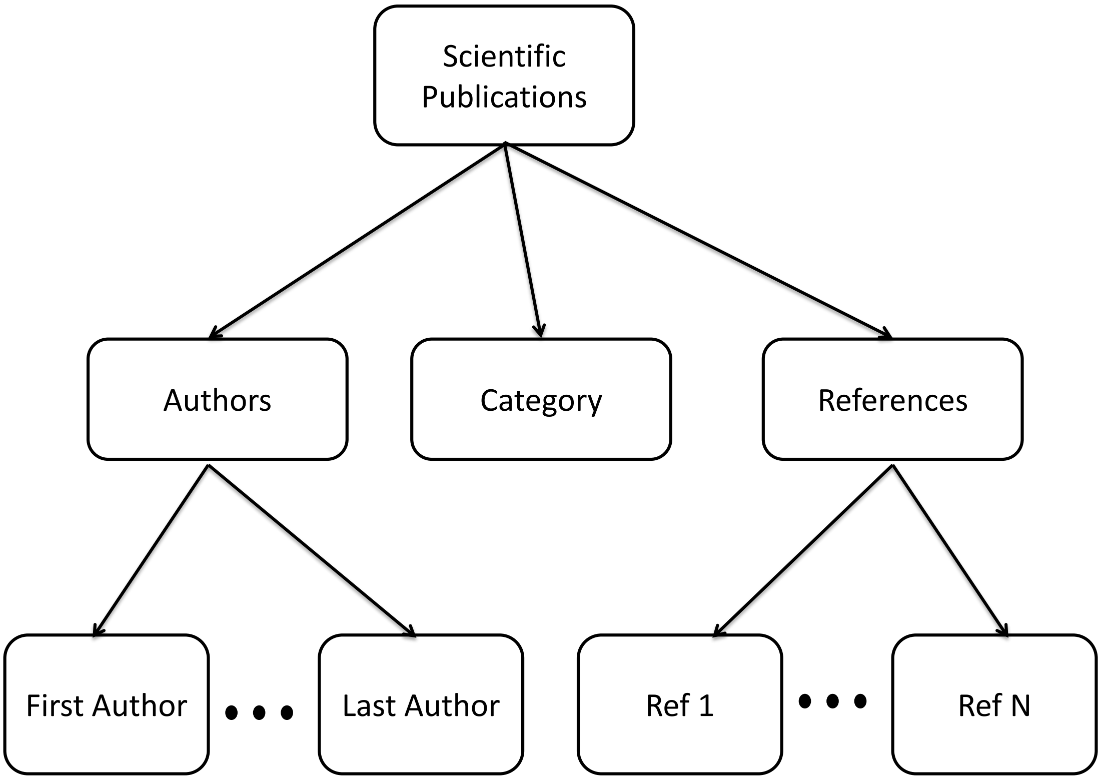
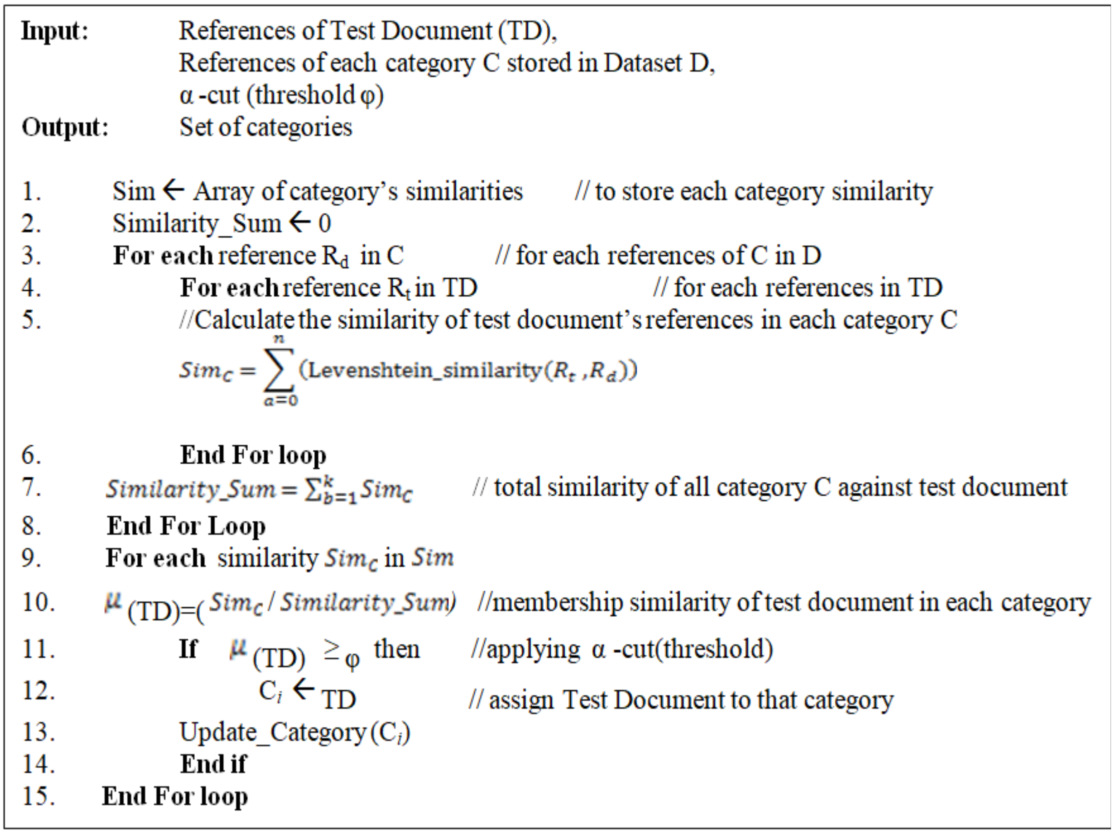


Figure 4.2: Extracted Metadata Information

and their references. When a user inputs a test document to identify its category, the system utilizes the stored TR pairs to recommend the possible appropriate category or categories.

4.1.3 Citation Based Category Identification (CBCI) Algorithm

The CBCI algorithm is an important concept of this proposed framework. The CBCI elaborates the idea of identifying category for a user provided research paper and is presented in Figure 4.3. In pre-processing step (Section 4.1.2), we have generated TR pairs and stored them in a database for the identification of relevant categories. When a new test article is received in a system, the metadata extractor extracts references from it. The extracted references and already stored TR pairs are assigned to the classifier module for further processing. The classifier module is based on the well-renowned Levenshtein similarity method [66]. This similarity method is widely utilized in the literature [67–69]. In the classifier module, test paper's references are matched with all the TR pairs. After the matching process, the membership Sim_c (System generated Category Weight (SC_W) termed as similarity from hereafter) of the test document is computed with respect to each category. We have also calculated the total similarity (Similarity_Sum) for all the categories. The next step is the computation of membership ($\mu(TD)$) (some weights or association of test document with each category) of the test document with respect to each category. After locating the membership ($\mu(TD)$), the next question is how to predict or assign the most relevant category for the test document? For this purpose, CBCI predicts the most relevant category or set of categories for the test document on the basis of its membership ($\mu(TD)$) in each category. This membership represents the strength of the resultant classification category generated by the proposed system. The higher membership increases the probability of assigning that category to the test documents. At this point, there are two possibilities to predict the most relevant category or categories for the test document. One is to select the top most category which has the highest membership among all categories and other possibility is to apply some "φ"-cut (threshold) to predict more than one categories. The process of threshold selection is performed by the domain expert after applying distinct experiments which are discussed in



```
Input:      References of Test Document (TD),
            References of each category C stored in Dataset D,
            α -cut (threshold φ)
Output:     Set of categories

    Sim ← Array of category's similarities      // to store each category similarity
    Similarity_Sum ← 0
    For each reference $R_d$ in C            // for each references of C in D
        For each reference $R_t$ in TD             // for each references in TD
        //Calculate the similarity of test document's references in each category C
        $Sim_C = \sum_{a=0}^{n} (\text{Levenshtein\_similarity}(R_t, R_d))$
        End For loop
    $Similarity\_Sum = \sum_{b=1}^{k} Sim_C$        // total similarity of all category C against test document
    End For Loop
    For each similarity $Sim_C$ in $Sim$
    $\mu$(TD)=($Sim_C$ / $Similarity\_Sum$)  //membership similarity of test document in each category
        If $\mu$(TD) ≥ φ then        //applying α -cut(threshold)
            $C_i$ ← TD               // assign Test Document to that category
        Update_Category($C_i$)
        End if
    End For loop
```

Figure 4.3: Citation Based Category Identification (CBCI) Algorithm

section 4.2.1. After the category or set of categories prediction, the next task is to update the knowledge-base (repository) of the proposed framework. For this task, CBCI updates TR pairs for the particular category or set of categories which are suggested to the input document. In this way, the knowledge-based repository is enriched to enhance the performance of our classification approach.

4.2 Results and Discussion

The implemented approach is implemented and evaluated on two different datasets, one is the Journal of Universal Computer Science (J.UCS) dataset [37] and another is an ACM dataset [18]. The detailed description of the results of these two datasets is presented in the following sections.

4.2.1 Threshold (α-cut) Tuning

The proposed CBCI algorithm (as presented above in the Figure 4.3) has performed multi-label document classification on the basis of a threshold value. This threshold value acts as an impediment for the prediction of categories. It is selected from the membership values computed by the CBCI algorithm. For a test paper, if the computed membership value for a category is either equal or higher than that threshold value, then that category is included in the set of predicted categories for a test paper. Different experiments are also performed to find out the optimum threshold value at which we may acquire the best results.
We have performed different experiments on J.UCS dataset (see Table 4.1) regarding the prediction of a set of categories on different threshold values. For instance, when we have matched test paper's references with all categories-wise references, the CBCI algorithm forecasts those categories (topics or classes) which contain a large number of references and the categories which have a low number of references were not predicted. Due to this, the efficiency of CBCI algorithm got decreased. After critically analyzing this problem, it has been identified; these results are due to class-imbalanced problem [70]. To resolve this issue, we have balanced/normalized all categories to a fixed number (1,000) references from each category. By doing this, CBCI algorithm returns significant results, by taking 1,000 references from each category for matching with the test paper's references and all extra references are discarded for comparison. The reason to select this

value is that we have 11 distinct root level categories as described in Table 4.2. Each category has different numbers of references. All categories encompass more than 1,000 references except categories A, E which contains approximately 1000 references. In category A and E, all references are selected and for remaining categories, 1000 references are selected from each category.

The selection of optimum threshold value is a challenging task for domain experts because the success of outcomes is dependent upon it. For threshold value selection, the comprehensive series of experiments is conducted. In the first attempt, we have performed experiments on a smaller data size comprises of 100 research papers. These 100 research papers are taken from the different 11 ACM categories (topics); for instance, "Information systems", " Computer Systems Organization", "Theory of Computing" etc. The results of these experiments yielded that membership values are significant in the range from 0.090 to 0.110 threshold. For validation, we have performed more experiments on different threshold values. In the first attempt, the threshold value of 0.100 is chosen; CBCI algorithm predicts reasonable numbers of categories against this threshold. In next phase, the threshold value is increased up to 0.105. The results at this threshold value are not as good as to threshold value of 0.100. As the threshold value increased from 0.105, the performance of CBCI algorithm became slightly poor. Similarly, in next phase, the threshold values are stepped down from the first experiment to 0.090. At this value, the CBCI algorithm has predicted the large number categories. For instance, if the original categories of the research paper are one or two, the CBCI algorithm predicts more than three categories. This behavior has adversely affected the outcomes of CBCI algorithm.

To discover the best threshold value, more experiments are performed by increasing threshold values from 0.090 to 0.110 with an increment of 0.001. For this purpose, we started the evaluation from a minimum threshold value, by analyzing the performance of the proposed system at threshold value of 0.080 and evaluated CBCI performance. The threshold values are gradually increased from 0.080 to 0.200. After different rounds of experiments, the significant results are achieved against threshold values ranges from 0.090 to 0.110. But the performance remained low from 0.080 to 0.09 due to the large number of predicted categories and also remained low from 0.110 to 0.2 due to very low numbers of predicted categories or even predicted noting at relatively higher threshold values. Hence, we have selected the range of membership values for each category from 0.090 to 0.110 as threshold values. Now the next important step is the identification of one optimal threshold value of this range. For this purpose, all the values in this range are applied on small dataset comprises of 100 research papers taken form J.UCS and ACM datasets. The evaluation parameters results are analyzed for different data sizes (research papers) like 10, 20, 30,. . . , 100 at different threshold values like 0.090, 0.091, 0.092,. . . , 0.110. This experiment is conducted to identify the most suitable threshold value at which the CBCI algorithm yields significant results. Another such experiment is conducted by increasing the number of dataset (i.e. 2.5 times bigger data size) and switching to J.UCS dataset. After this, the best threshold value is compared with the first type of experiments on another 2.5 times bigger data size (ACM research papers). These multiple rounds of experiments are performed to accurately identify the most optimal threshold value. The detailed descriptions of these three types of experiments are presented as follows:

For the first experiment, we have used metadata (Reference's Section) of 100 research papers belonging to diversified topics from both J.UCS and ACM datasets. There are some research papers which belong to single-label, but most of them belong to multiple classes. The CBCI algorithm has been applied after the preprocessing steps for the multi-label document classification. The experiments on smaller dataset are conducted is to identify the most suitable threshold value at which the proposed CBCI algorithm yields significant numbers of results and also to select the range of threshold values for further experiments on two big and diversified datasets.

We have evaluated the results of 100 research papers on threshold range from 0.09 to 0.11 in such a way that; first we have computed the evaluation parameters values on 10 research papers on this threshold value range and identified that at threshold value 0.097; the CBCI algorithm produces the best results. In the next experiment, the number of research papers are expanded by adding 10 more research papers to validate the previous best threshold value and surprisingly the same threshold value is 0.097 is reported here as well. Similarly, in all rounds the data size is increased by adding 10 research papers every time to validate the previous best reported threshold values on threshold value range from 0.09 to 0.11. We have examined that the best threshold value is increasing gradually by increasing the number of research papers. When we increase the number of papers, at 50, we achieved the threshold value of 0.1 which remained constant or negligible change has been occurred while increasing the number of papers from 50 to 100. Each set of research papers are evaluated on threshold range from 0.09 to 0.11 and analyzed the best threshold value for each set of research papers at which we have achieved the best accuracy.

In the Figure 4.4, we have shown each set of research papers on X-axis and on the Y-axis, the only best threshold values is presented on which we achieved the best accuracy for each set of research papers. Hence from the first type of experiments on 100 research papers our proposed model achieved best results at threshold value of 0.1 and the results of all the best threshold values on which we achieved the best results for each 10 set of research papers are presented in the Figure 4.5. In the above experiments for 100 research papers, it can be observed that the best threshold value remained 0.100. After this threshold value, the accuracy, precision, recall, and F-measure started to drop by increasing the number of research papers. For further confirmation of the predicted best threshold value (0.100) for the first type of experiments on a smaller dataset of 100 research papers, more experiments are performed to evaluate this best threshold value (0.100) on two big data sizes i.e. 2.5 times bigger and diversified data sizes as compared to the above first type of experiments. The detailed description and statistics of these big datasets are presented in the following sections.

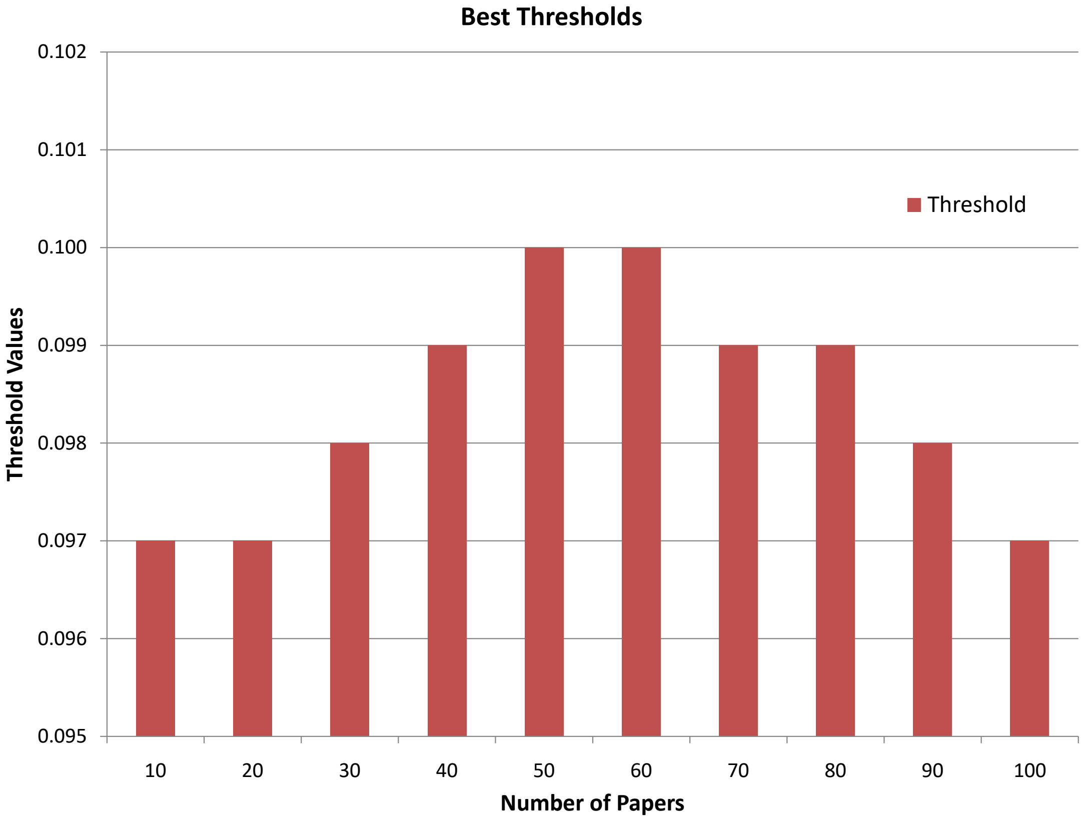


FIGURE 4.4: Number of Papers Vs Threshold Values

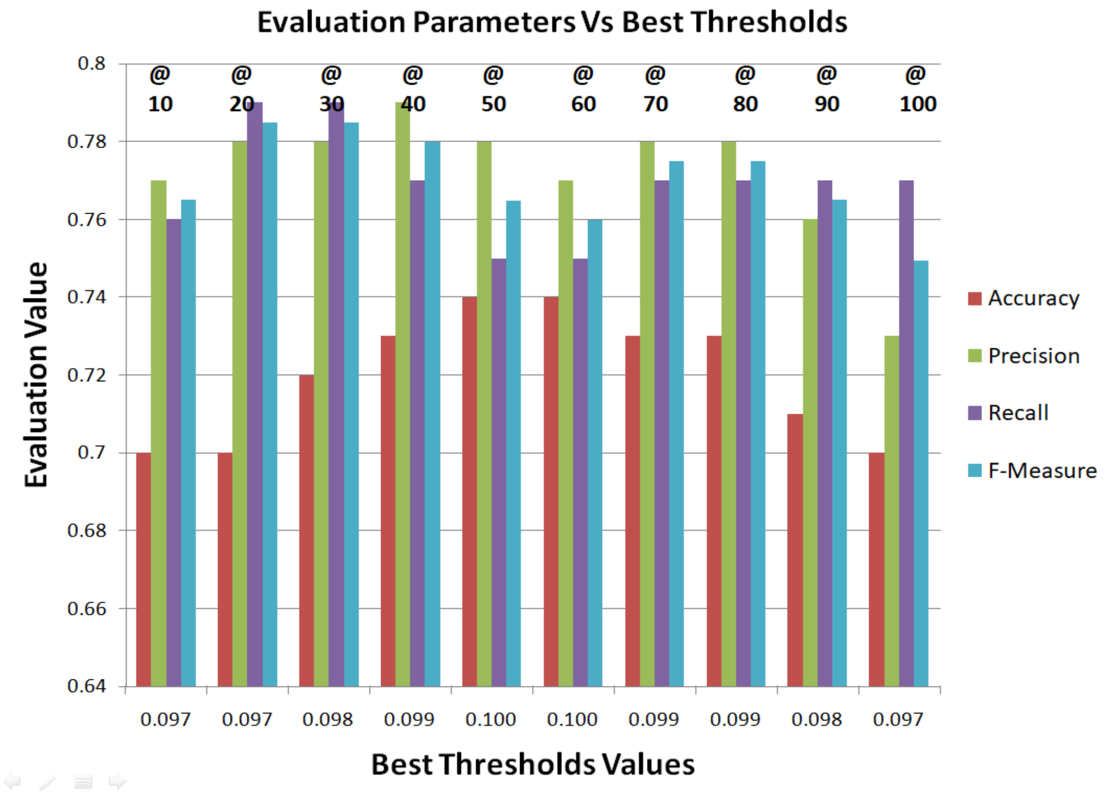


FIGURE 4.5: Evaluation Parameters Vs Threshold values

TABLE 4.1: Features of the J.UCS dataset

Features	Values
Total Number of References	16,404
Average Number of References in Each Paper	11
Total Papers with References Used	771
Distinct References Used	15,385
Topic Reference Pairs	48,175

4.2.2 Evaluation of Threshold on J.UCS Dataset

The Journal of Universal Computer Science (J.UCS) contains research publications from the diversified domain of Computer Science. The statistics for J.UCS dataset are presented in Table 4.1. These research papers contain 16,404 total numbers of references. One research paper may belong to more than one category as described in Table 3.2 that there are significant numbers of research papers which belong to more than one category. This dataset contains 48,175 total numbers of Topic References (TR) pairs and references in each category are shown in the Table 4.2. Class imbalanced problem is problem is already discussed for J.UCS dataset in the above section. There are 1,460 research papers on J.UCS

Table 4.2: J.UCS: References in each Category

Category	References	Category	References
A	767	G	1,420
B	1,053	H	11,883
C	3,175	I	6,141
D	9,816	J	1,657
E	853	K	3,750
F	7,304		

dataset, from these 771 papers' references are stored in the dataset that make pairs with the topics (categories). Total references stored in dataset are 16,404 and total distinct references used for generating TR pairs are 15,385. From these references and category information, we have generated 48,175 TR pairs and stored them in the database for classification of the research papers.

For the evaluation of experiments on above benchmark threshold value (0.100), we have performed our experiments on J.UCS dataset (250 research papers) to find out the performance of our CBCI algorithm and to make sure whether the 0.1 threshold remains the best one or not for this dataset too. For this, we have also applied the same standard evaluation measures proposed by Godbole and Sarawagi [62] for the multi-label classification as described in Chapter 3 (Section 3.3.1 i.e., accuracy (Eq. 3.2), precision (Eq. 3.3), recall (Eq. 3.4) and F-measure (Eq. 3.5) are applied.). The in-depth analysis of the results revealed that at threshold value 0.1 for J.UCS dataset have achieved F-measure value of 0.77 which is much closer to the F-measure value 0.76 achieved for the first type of experiments on 100 research papers. The results of J.UCS dataset are further investigated on different threshold values like 0.090, 095, 0100, 0.105 and 0.110. The detailed description and analysis of these experiments and their results are given below.

In Table 4.1, it has been described already that in J.UCS dataset, there are 1,460 research papers and from these research papers, only 1,147 research papers are evaluated since only these articles have the author's provided categories list. Therefore, from these 1,147 research papers, only 771 research papers contain references. There are 16,404 total references in the dataset and there are 15,385 total distinct references in the dataset through which 48,175 TR pairs are generated in the pre-processing step (Section 4.1.2). For testing, first 250 research papers and their references are selected from the J.UCS dataset for the evaluation of our approach. After the completion of pre-processing step, these test papers' references and category-wise stored research papers references are given as input

Table 4.3: J.UCS Dataset Results

Threshold	**Accuracy**	**Precision**	**Recall**	**F-Measure**
0.090	0.63	0.65	0.86	0.74
0.095	0.67	0.69	0.84	0.76
0.100	0.71	0.73	0.81	0.77
0.105	0.74	0.78	0.81	0.79
0.110	0.69	0.77	0.75	0.76

to the Citation Based Category Identification (CBCI) algorithm for the prediction of research papers categories. In comparison, test paper's references are not included while matching with all categories-wise references stored in the dataset. The working of CBCI algorithm is presented in the Figure 4.3. CBCI algorithm predicts a set of categories on the basis of different threshold values as already described that the experiments are performed on different threshold values. Results for J.UCS dataset of metadata (Reference's Section) are presented in Table 4.3. For the performance evaluation of these predicted results, same previously utilized evaluation parameters are applied, as described in Section 3.3.1. In Figure 4.6, J.UCS dataset results are presented according to the evaluation parameters for the multi-label document classification by exploiting the reference's section of the research papers. The evaluation parameters values are presented on Y-axis and threshold values on X-axis. The in-depth analysis of results of J.UCS dataset for metadata (*Reference's Section*), the findings are as listed below:

- The most optimum threshold value of these results is 0.105, at this threshold value CBCI algorithm yields significantly better results for all evaluation parameters except for recall.
- When we gradually increase the threshold value from the minimum value towards 0.105, the performance of CBCI algorithm becomes poor due to very low numbers of predicted categories at high threshold value.
- Similarly, when we gradually decrease the threshold value from 0.105, the performance of CBCI algorithm suffers due to the large number of predicted categories at the low threshold values.

We have performed our experiments on the J.UCS dataset and performance evaluation of these experimental results is conducted by using the evaluation parameters proposed by [62] for the multi-label document classification. These experimental

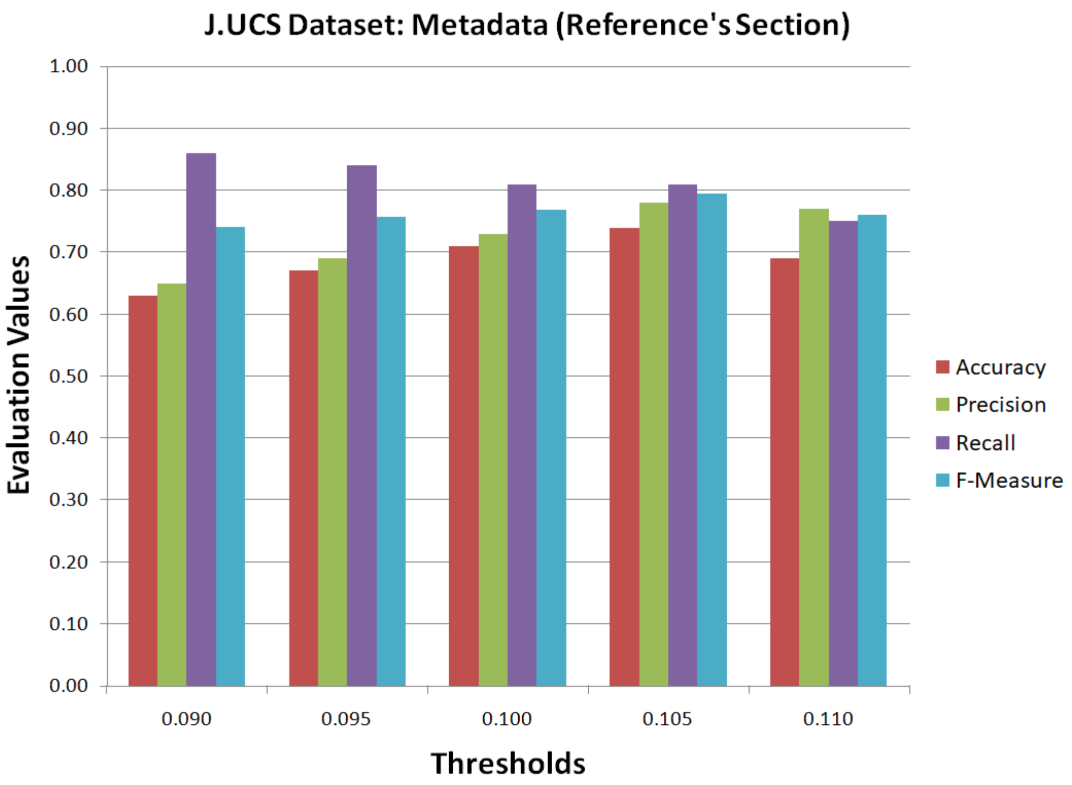


FIGURE 4.6: J.UCS: Evaluation Parameters Vs Threshold Values for *Reference's Section*

results are thoroughly examined by applying different threshold values to CBCI algorithm. The algorithm yields different results at different threshold values. CBCI algorithm performance is significantly better at threshold value 0.105 as shown in the Figure 4.6 for the J.UCS dataset. At this threshold value, the CBCI algorithm achieved F-measure value of 0.79 which is much closer to the benchmark threshold value's (0.100) F-measure (0.77). The error rate for this best threshold value (0.105) for J.UCS dataset as compared to the benchmark threshold value (0.100) is 0.02. Since, the ideal performance of CBCI algorithm is reported at the threshold value of 0.100, therefore, this value is contemplated as the benchmark value. The benchmark threshold value (0.100) is further investigated for another big and diversified dataset (ACM dataset). The detailed description and statistics of this big dataset (ACM dataset) are presented below.

Table 4.4: Features of the ACM dataset

Features	**Values**
Total Number of References	196,138
Average Number of References in Each Paper	18
Total Papers with References Used	3,939
Distinct References Used	65,442
Topic Reference (TR) Pairs	1,59,681

Table 4.5: ACM: References in each Category

Category	**References**	**Category**	**References**
A	342	G	2,907
B	6,683	H	32,339
C	21,136	I	23,838
D	47,597	J	1,123
E	723	K	15,319
F	7,674		

4.2.3 4.2.3 Evaluation of Threshold on ACM Dataset

The Association for Computing Machinery (ACM) dataset [18] contains 86,116 research publications from different workshops, conferences and journals. It contains 11 distinct categories at the top level of the ACM computing classification system (CSS). The statistics for ACM dataset are presented in Table 4.4. From these research papers, we have randomly picked 5,403 research papers and have extracted their references from the web links which are provided in the dataset. There are 96,138 references in 5,403 research papers. One research paper may belong to more than one category as described in Table 3.2 that there are significant numbers of research papers which belong to more than one category. There are 86, 116 research papers in ACM dataset, from which 3,939 research paper's references are stored in the dataset and Topic Reference pairs are generated. Total references stored in dataset are 96,138 and total distinct references used for generating TR pairs are 65,442. From these references and category information, 1,59,681 TR pairs are generated and stored in the database which is used for the classification of the research papers (documents).

The class-imbalanced problem is reported for ACM dataset as well which is resolved by selecting the same number of references i.e. 1,000 references from each category. By doing this, CBCI algorithm returns significant results to find similarity for the test paper's references. Reason to select this value is that we have 11 distinct root levels of categories at ACM hierarchy as described in Table 4.5.

Table 4.6: ACM Dataset Results

Threshold	Accuracy	Precision	Recall	F-Measure
0.090	0.70	0.69	0.83	0.75
0.095	0.74	0.71	0.79	0.75
0.100	0.71	0.70	0.76	0.73
0.105	0.70	0.68	0.73	0.70
0.110	0.67	0.65	0.70	0.67

Each category has different numbers of references. All categories contain more than 1,000 references except categories A, E. In category A and E, all references are selected and from all other categories, 1000 references are selected from each category.

To evaluate the benchmark threshold value (0.100) at ACM dataset, multiple rounds of experiments are conducted. Same as J.UCS experiments, the standard evaluation measure, i.e., accuracy (Eq. 3.2), precision (Eq. 3.3), recall (Eq. 3.4) and F-measure (Eq. 3.5) are applied. The results of this dataset are examined thoroughly; the results yielded that the CBCI algorithm's performance is significant by achieving F-measure value of 0.73 for benchmark threshold value. Same behavior as of smaller and bigger dataset (J.UCS) is reported here as well. The results of ACM dataset are examined further on different threshold values like 0.090, 095, 0100, 0.105 and 0.110. The detailed description and analysis of these experiments and their results are as given below:

It has been described already (see Table 4.1) that in ACM dataset, there are 86,116 research papers and from these, only 54,994 research papers are evaluated because only these articles have author's provided categories. Therefore, from these 54,994 research papers, only 3939 research papers contain references. There are 96,138 total references and 65,442 total distinct references. Total 1,59,681 TR pairs are generated in the preprocessing step (Section 4.1.2). For the testing phase, first 250 research papers and their references are selected from ACM dataset. After completion of pre-processing step, these test paper's references and category-wise stored research papers references are given as an input to CBCI algorithm for the prediction of research papers categories. In comparison, test paper's references are not included while matching with all categories-wise references stored in the dataset. For the performance evaluation of these predicted results, same evaluation parameters are utilized as described in Section 3.3.1. In Figure 4.7, ACM dataset results are presented according to the evaluation parameters for the multi-label document classification by exploiting the reference's section of the research papers. The evaluation parameters values are presented on Y-axis and threshold values on X-axis. Based on the in-depth analysis of ACM dataset results for metadata (*Reference's Section*), the findings are listed below:

- The most optimum threshold value for these results is 0.095, at this threshold value CBCI algorithm yields significantly better results for all evaluation parameters except for recall.
- When the threshold value is gradually increased from 0.095, the performance of CBCI algorithm becomes poor due to the number of predicted categories at higher threshold values.
- Similarly, when the threshold value is gradually decreased from 0.095, the performance of CBCI algorithm becomes poor due to the large number of predicted categories against low threshold values.

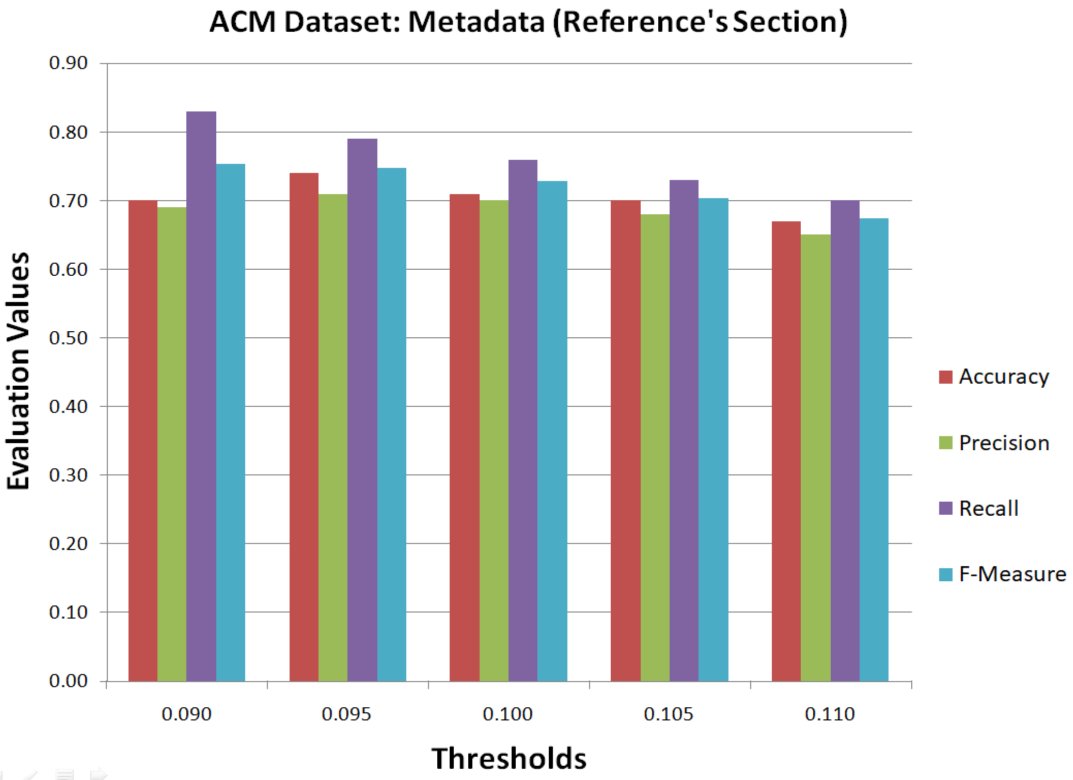


FIGURE 4.7: ACM: Evaluation Parameters Vs Threshold Values for *Reference's Section*

The experiments are performed in the ACM dataset and performance evaluation of these experimental results is conducted by harnessing the evaluation parameters proposed by Godbole and Sarawagi [62]. These experimental results are critically analyzed by applying different threshold values to our CBCI algorithm. It yields different results at different threshold values. The CBCI algorithm's performance is significantly better at threshold value 0.095 as presented in the Figure 4.7 and achieved F-measure value of 0.75 which is much closer to the benchmark threshold value (0.100) with a negligible error rate of 0.02.

Consequently, three different types of experiments are performed on a small dataset (100 research papers) and two large datasets (2.5 times bigger). Above experimental results render the most optimum threshold value is almost same for all both dataset (small and large) with an error rate of 0.02. By doing comprehensive experiments on different data sizes, it is identified that at threshold value, 0.100; the proposed model yields significant numbers of results for smaller as well as for bigger datasets by exploiting only metadata (*Reference's Section*) of the research papers.

4.3 Single-label Document Classification

As discussed earlier, the proposed framework is also intended to perform single-label document classification. For evaluation, same two diversified datasets (J.UCS and ACM) are harnessed to perform single-label classification. For single-label classification, the proposed CBCI algorithm predicts only that category which has the highest membership value among all other categories. If there is a tie among more than one category with their highest membership value then all those categories are considered. For single-label document classification there is no need of threshold value as it is a binary decision whether a research paper is correctly classified or not. For evaluation, the well-known confusion matrix is utilized which is discussed in Chapter 3 (Figure **??**).

The experimental results have been evaluated for 250 research papers from the J.UCS dataset. We have assumed that the categories are accurately annotated by the authors of research papers. The experiments are conducted on all 250 distinct research papers' category pairs. In the Figure 4.8, the evaluation parameters are plotted on the X-axis and their values are plotted on Y-axis. Based on scrutinization of the obtained results for all evaluation parameters with respect to the different metadata values, the observed outcomes are listed below:

- The CBCI algorithm performs well against all the evaluation parameters for metadata (*Reference's Section*) and almost 0.88 times it correctly classifies the research papers into single-label.
- We have assumed that the annotated categories are correct, that's why precision of CBCI algorithm is 100% as all predicted categories are correct and there is no False Positive (FP).
- The proposed approach's accuracy and recall have the same values because there is no True Negative (TN) value in the confusion matrix.

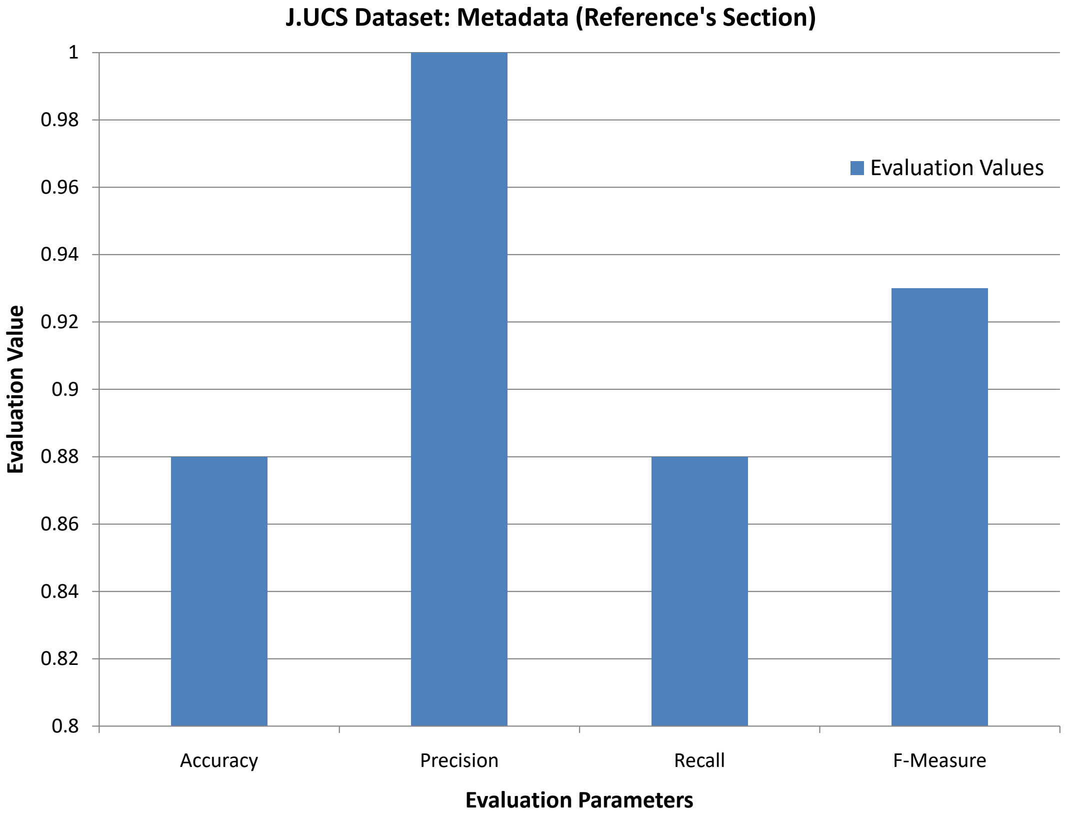


Figure 4.8: J.UCS Dataset: Results for Evaluation Parameters

Similarly, the experiments are performed on ACM dataset and by selecting 250 research papers from ACM dataset. The same assumption is followed here as well, that the categories annotated by the authors of these research papers are correct. The experiments are performed on all 250 distinct research papers' category pairs. In the Figure 4.9, evaluation parameters are plotted on X-axis and their values are plotted on Y-axis. Following points are observed based on the detailed analysis of results.

- The CBCI algorithm performs well against all the evaluation parameters for metadata (*Reference's Section*) and almost 0.84 times it correctly classifies the research papers into the single-label.
- We have assumed that the annotated categories are correct, that's why precision of our CBCI algorithm is 100% as all predicted categories are correct and there is no False Positive (FP).
- The proposed approach's accuracy and recall have the same values because there is no True Negative (TN) value in the confusion matrix.

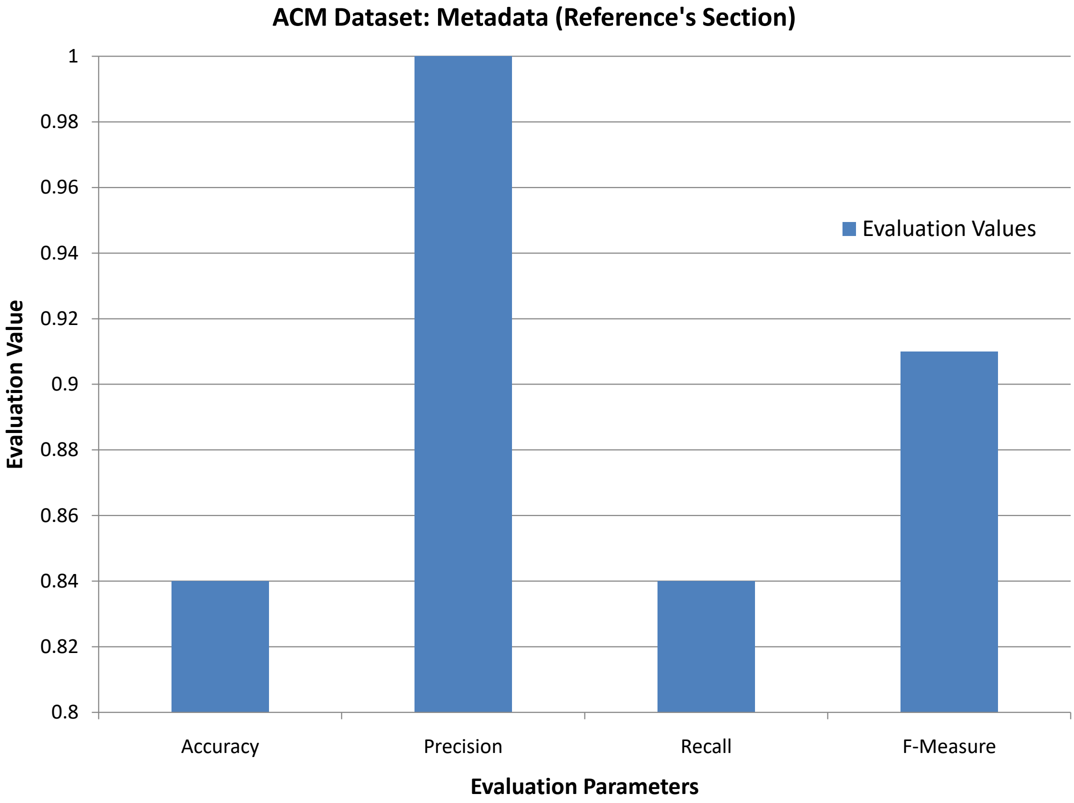


FIGURE 4.9: ACM Dataset: Results for Evaluation Parameters

4.4 Summary

The digital corpora of research publications is getting doubled every five years and demands an effective mechanism to return most relevant results against user posed query. The document classification community is aimed at producing effective techniques to classify research articles. The classification process is expanded into two categories (1) Single-label and (2) Multi-label. The single label classification requires a comparatively small set of features than multi-label classification. We argue that a research paper can belong to multiple categories due to diversified linkage between multiple topics and domains. The multi-label document classification is crucial yet demanding issue due to association of research article to multiple categories. In this chapter, a novel approach for multi-label document classification named as Category Based Citation Identification (CBCI) is presented. The classification is performed by applying freely available metadata due to its free availability and potential. The proposed system not only performs multi-label classification, but also single-label classification. The system takes article as an input and assigns a new category or categories on the basis of already stored TR pairs. Different experiments are performed by exploiting only metadata (Reference's Section) of the research articles. Two diversified datasets ACM and J.UCS are utilized. Topic-Reference (TR) pairs are generated from these datasets and stored in the database to assign categories for input document.

The classification is depended upon threshold value, to find out the optimum threshold value for CBCI algorithm; different rounds of experiments are conducted on both sets. At each round, the value of data size is set to distinct values in order to discover the benchmark threshold value. The experimental results depicted that the proposed model achieved good results at threshold value 0.1 for 100 research papers. The results of this threshold value are much closed to the best results for 2.5 times bigger datasets (J.UCS and ACM) with a very low error rate of 0.02. Consequently, the optimum threshold value is 0.1 at which proposed model yields significant numbers of results by exploiting only metadata (*Reference's Section*) of the research papers and performs multi-label document classification. Similarly, in case of single-label classification, the proposed model achieved significant results. The results for both single-label and multi-label document classification at threshold value 0.1 are compared with the state-of-the-art classification approaches in the Chapter 5.

However, the process of matching test paper's references with all stored TR pairs is a prolonged process due to the large number of TR pairs. The matching efficiency could be increased by adding some heuristics like indexing the references according to their length. Organizing the references in the order of their length makes the matching process efficient by comparing each of the references of test paper with a similar length of references in the database. For example, if the length of the reference in test paper is L, this reference will be compared with only the references having a length from $L - E$ to $L + E$, where E is the error length in the references.

Chapter 5

Evaluation and Comparative Analysis

This dissertation has presented two novel approaches in Chapter 3 and Chapter 4 to perform multi-label classification of research articles from Computer Science domain by relying fully on freely available metadata based parameters. The pivot of this chapter focuses on evaluation and comparisons of proposed mechanisms with existing techniques to investigate their status in the current state-of-the-art. The proposed approaches have employed freely available metadata in a best possible way. Different metadata parameters are utilized in both approaches.

1. Multi-label Document Classification using Papers' Metadata (*Title* & *Keywords*).
2. Multi-label Document Classification based on *Papers' References*.

The schemes have been evaluated on two diversified datasets, (1) J.UCS [37] and (2) ACM [18].Their detailed descriptions and results have been elaborated in the previous two chapters. This chapter presents the comparison and critical evaluation of the achieved results. To justify the comparisons, there should be certain similarities between proposed and existing approaches. We have performed in-depth analysis of literature. As best of our knowledge, there exists no scheme which performs multi-label classification by using only freely available metadata. Those which have utilized metadata of the research papers only perform single-label classification. The document classification community is dominant with content based approaches; therefore, there is an extensive amount of these schemes in literature. Almost all of these content based schemes have performed single-label classification. We found only one approach which performs multi-label classification, but depends on content based features. The basic aim of this dissertation is to examine that to what extent metadata can be employed to classify research papers into multi-labels. Therefore, to compare proposed approaches with state-of-the-art approaches, we had two options which are described below both of these have been chosen to comprehensively evaluate the proposed approaches:

- First, we have compared state-of-the-art approaches which perform multi-label classification by using the content of the research papers. However; we have also performed multi-label classification, but using freely available metadata.
- Secondly, we have compared single-label classification state-of-the-art approaches which perform single-label classification by using the metadata of the research papers on our both datasets. We have also performed our experiments to classify papers into a single class by using the metadata of the research papers as described in the end of both Chapter 3 and Chapter 4.

Therefore, we have compared and evaluated the proposed model's experimental results with the state-of-the-art approaches which perform single-label classification as well as those state-of-the-art approaches which perform multi-label classification. The detailed description of these state-of-the-art approaches is presented in the Table 5.1. In the Table 5.1, there are three approaches that utilized metadata of the documents and performed single-label classification. Similarly, there are three approaches which used content of the documents and performed multi-label classification. There are some other approaches (as described in the Chapter 2) which used content of the documents and performed single-label classification. These approaches are not included in the Table 5.1 because the focus of this dissertation is on two essences; one is metadata of the documents and other one is the multi-label classification. As we have exploited metadata from the scientific documents, therefore, we have compared and evaluated our proposed approaches with only those state-of-the-art approaches which have also utilized the scientific documents and have discarded all other approaches which have employed other types of documents like web pages, news, email, genes etc.

In Table 5.1, an approach proposed by Flynn [35] used heterogeneous Defense Technical Information Center (DTIC) collection, documents are diverse, including slides, from presentations, public laws, acts of Congress, conference proceeding, scientific documents and PhDs dissertation. There are only two approaches which have utilized the scientific documents for the classification. One approach was proposed by Khor and Ting [14] which used metadata of the scientific documents and performed single-label classification and other approach was proposed by Santos and Rodrigues [18] which used content of the scientific documents and performed multi-label classification. Hence we have compared and evaluated our proposed approaches with only these two approaches.

Table 5.1: State-of-the-art Approaches

S.No.	Approaches	Type of Data	Classification Type	No. of Classes	Dataset	Algorithm / Methodolgy	Evaluation Parameters	Results
1	Flynn, 2014	Metadata	Single-Class	99	2000 Documents	Independent Document Model Framework	Precision, Recall, F-Measure	Precision (0.79), Recall (0.81), F-Measure (0.79)
2	Khor and Ting, 2006	Metadata	Single-Class	4	400 Documents	Baysian Network (BN), Naive Bayes (NB), Baysian Network Learner (BNL)	Accuracy	Accuracy (BN,0.84; NB, 0.83; BNL, 0.76)
3	Zhang et al., 2004	Metadata	Single-Class	11	30,000 Features	Genetic Programming (GP)	Accuracy	Accuracy (0.61)
4	Santos and Rodrigues., 2009	Content	Multi-Class	11	5,000 and 10,000 Documents	Binary Relevence (BR), Label Power (LP), Multi-Label kNN (MLkNN)	Accuracy	Accuracy (0.88)
5	Lijuan, 2008	Content	Multi-Class	WIPO-alpha (8), Newsgroup (5), OHSUMED (15), ENZYME (236)	Synthetic Data, WIPO-alpha, Newsgroup (1000),OHSUMED (54,708), ENZYME (9,455)	Hierarchical SVM, Hierarchical Perception	Accuracy, Precision	Accuracy (0.94), Precision (0.89)
6	Wang and Desai, 2007	Content	Multi-Class	6	45,000 Features	Naive Bayes, Centroid	Accuracy	Accuracy (0.61)

First, the proposed approaches are compared with the content based multi-label classification approach proposed by Santos and Rodrigues [18]. Secondly, the proposed approaches are compared with the metadata based single-label classification scheme proposed by Khor and Tang [14]. The detailed descriptions of these both comparisons are elaborated in the following section.

5.1 Comparison with Multi-label Document Classification Approach

Multi-label document classification approach is proposed by Santos and Rodrigues [18] which utilizes the content of scientific documents. In this approach, scientific documents are extracted from the ACM digital library and multi-label classification is performed by using the content of these scientific documents. This system has achieved an accuracy of 0.88 by using the Binary Relevance (BR) classifier.

5.1.1 Comparison on ACM Dataset

We are interested to scrutinize that to what extent a remarkable accuracy can be achieved by using metadata based features instead of content based features. For the experiments, the metadata parameters such as *Title* and *Keywords* are extracted from the research papers to classify research papers into one or more than one category. The detailed descriptions of this proposed approach and its results are presented in Chapter 3. The experiments are performed on the same ACM dataset provided by Santos and Rodrigues [18] and found the optimal threshold value (0.2) at which our proposed approach produces significant results. The exploitation of metadata (*Reference's Section*) is introduced for the first time to classify research papers into one or more than one category. The detailed description of this novel approach and its results are presented in Chapter 4. Same as *Title* and *Keywords* based scheme, the experiments are performed on the ACM dataset provided by Santos and Rodrigues [18] and the most optimal threshold value (0.1) is identified at which the proposed approach produces significant results. The comparison of these multi-label proposed approaches' results at optimal threshold value with the multi-label approach proposed by the Santos [18] is presented in the Figure 5.1. It can be seen that the proposed approach achieved same accuracy as of the Santos scheme by exploiting the metadata (*Title*). Moreover, remarkable results are reported by exploiting metadata (*Keywords*), metadata (*Title & Keywords*) and metadata (*Reference's Section*). Apart from these results, we have

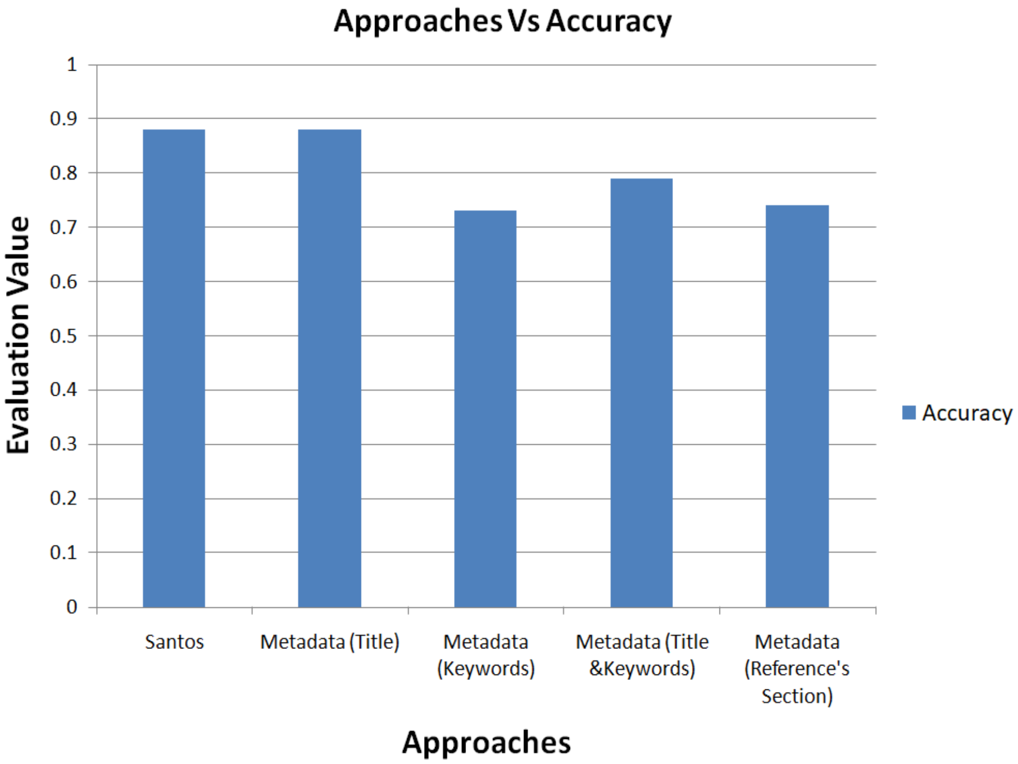


FIGURE 5.1: Comparison with State-of-the-art approaches

also evaluated ACM dataset on the classifier used by Santos and Rodrigues [18] such as Binary Relevance (BR), Label Powerset (LP) or Label Combination (LC) [71, 72] and some others multi-label classifiers such as Bayesian Classifier Chain (BCC) [34], Expect Maximization (EM) [73] and Hierarchical Classifier (HASEL) [54] provided by MEKA, an extension to WEKA open source tool. BCC classifier is applied in combination with J48 [74] and Naïve Bayes [75] classifiers. All others classifiers are applied in combination with J48 classifier, and function Sequential Minimal Optimization (SMO) [76]. For this purpose, total 10,000 numbers of instances are used for metadata (*Title*) and metadata (*Title & Keywords*) experiments. We have used Gain Ratio as a feature selection technique to perform the experiments. These features are extracted from the metadata (*Title*) and metadata (*Title & Keywords*) of the research papers. For experimental results, we have used all features from these instances and have performed pre-processing on the dataset. For evaluation, default settings are applied for each classifier, 90%

instances are considered for training and 10% instances are considered for testing. In the Figure 5.2, multi-label classifiers' results are compared with the results of proposed model for metadata (*Title*) by plotting these classifiers on X-axis and evaluation parameters values on the Y-axis. It can be seen that the performance of proposed model is quite good as compared to the results of all other multi-label classifiers. Moreover, the performances of all other multi-label classifiers are almost consistent for this particular ACM dataset. We have also evaluated different state-of-the-art multi-label classifiers for metadata (*Title* & *Keywords*) of research papers in ACM dataset. Default settings for all multi-label classifiers are used as we have used for metadata (*Title*) as described above. The multi-label classifiers results are compared with the results of the proposed model at optimal threshold value (0.2) which is presented in the Figure 5.3.In figure, the classifier name is plotted on X-axis and parameters values are plotted on Y-axis. It can be seen that the performance of proposed model is relatively good as compared to the results of all other multi-label classifiers. The performance of the state-of-the-art multi-label classifiers is also consistent and among these classifiers Label Combination (LC) with Sequential Minimal Optimization (SMO) performance is remarkable as

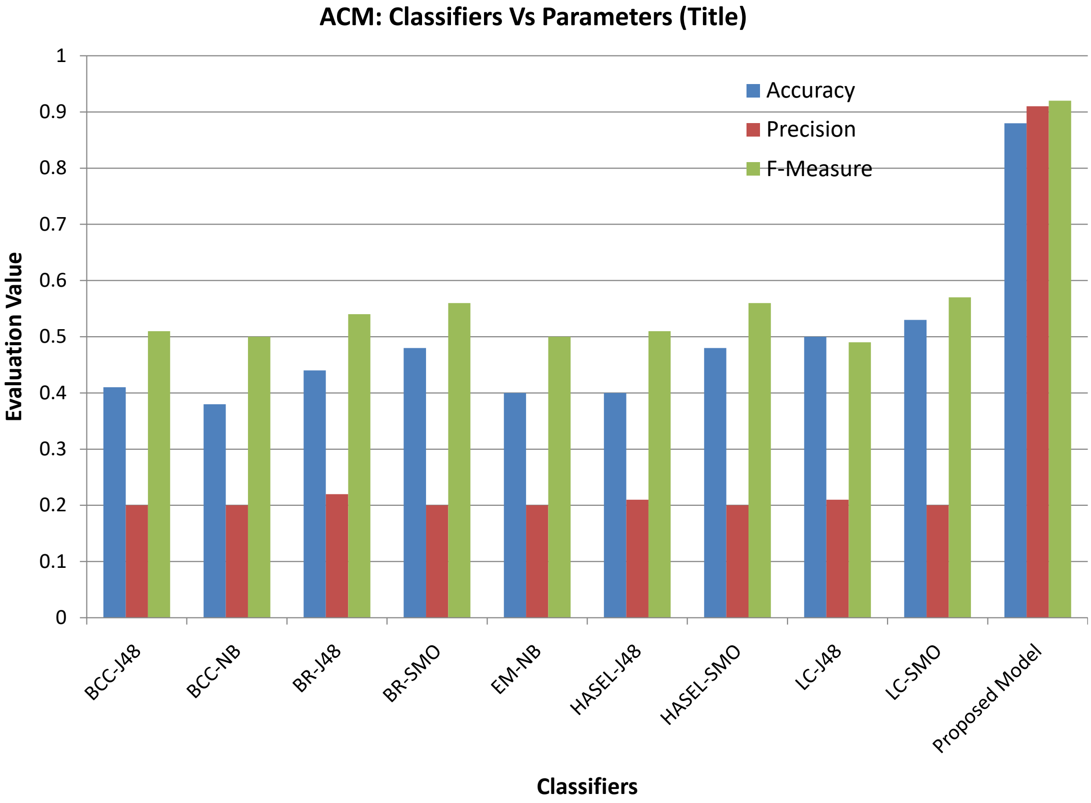


FIGURE 5.2: Multi-Label Classifiers Vs Evaluation Parameters (*Title*)

compared to the others state-of-the-art multi-label classifiers. Similarly, we have

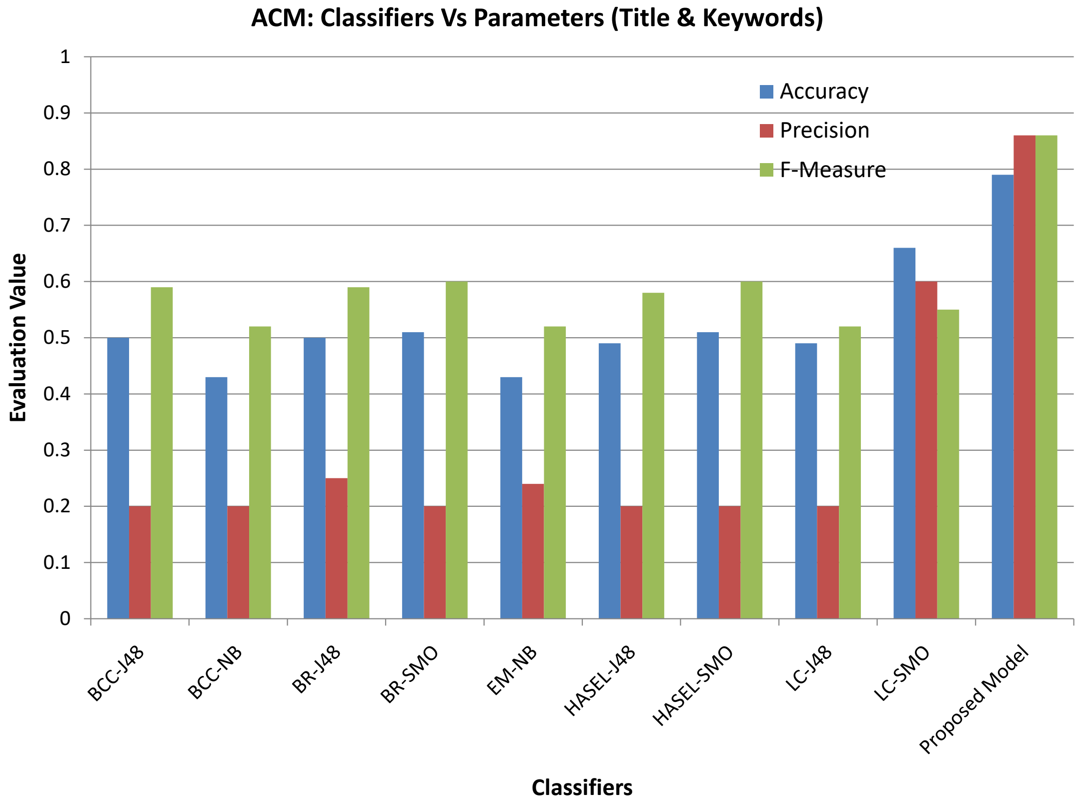


Figure 5.3: Multi-Label Classifiers Vs Evaluation Parameters (*Title & Keywords*)

also tested references strings from ACM dataset on different multi-label classifiers. The critical analysis of the state-of-the-art classifiers, their results and comparison with the proposed approach's results are presented below.

For the evaluation, the experiments are performed on ACM dataset to analyze the performance of results. The experiments are performed by using state-of-the-art multi-label classifiers as mentioned above on the Topic's References (TR) pairs. All these classifiers are applied with the Naïve Bayes Classifier. For these experiments, there are 4,000 total numbers of instances from which 90% are used for training and the remaining ones for testing. The default settings are applied for all state-of-the-art multi-label classifiers and computed the evaluation parameter values. These state-of-the-art classifiers results are evaluated against the proposed model results. In the Figure 5.4, the classifier name is plotted on X-axis and the evaluation parameters values are plotted on Y-axis. Different points are observed based on the critical analysis of state-of-the-art multi-label classifiers on ACM dataset for the metadata (*Reference's Section*).

- The performance of the proposed model is comparatively good as compared to the state-of-the-art multi-label classifiers results.
- The performance of Hierarchical Classifier (HASEL) is comparatively low as compared to other multi-label classifiers. The detailed results of state-of-the-art multi-label classifiers for metadata (*Reference's Section*) on ACM dataset are shown in Figure 5.4.

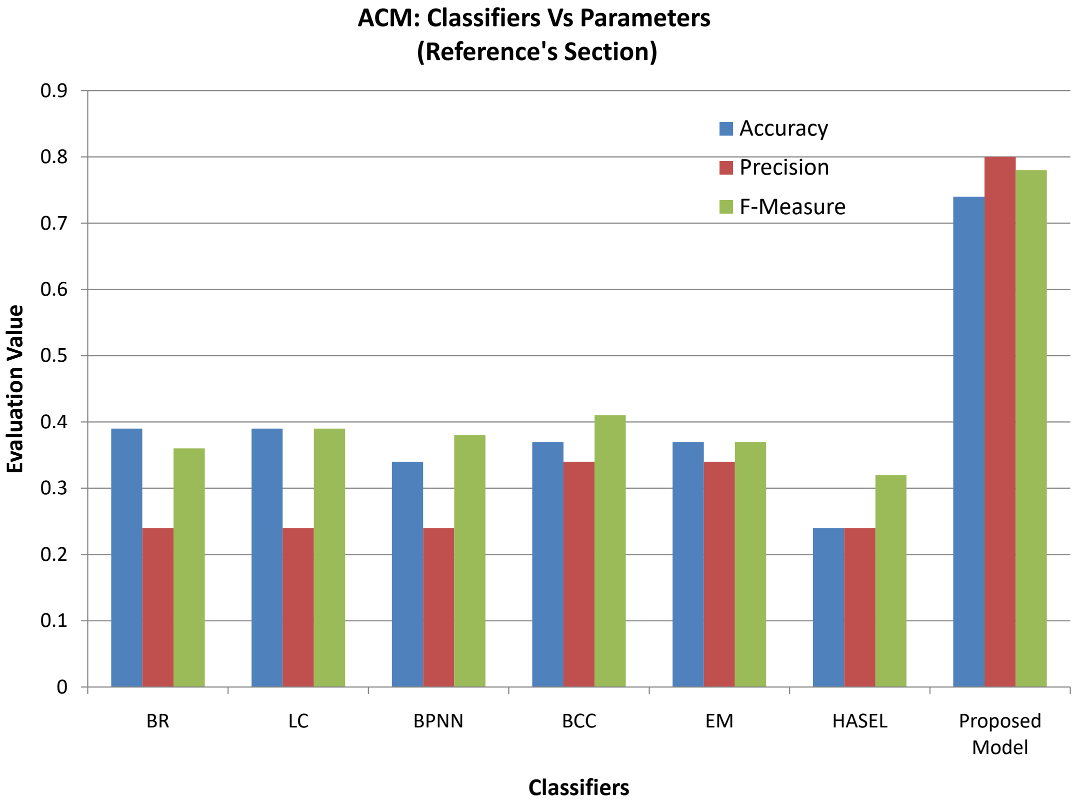


Figure 5.4: ACM: Multi-Label Classifiers Vs Evaluation Parameters (*Reference's Section*)

5.1.2 Comparison on J.UCS Dataset

Similarly, the experiments are also performed on another diversified J.UCS dataset which contains scientific documents from the different Computer Science domains. Different state-of-the-art multi-label classifiers (as described above) are evaluated on the J.UCS dataset. For evaluation, the metadata (*Title*) of all the research papers (instances) is extracted, default settings are applied for each classifier, 90% instances are taken for training and 10% instances are taken for testing. These multi-label classifiers results are compared with the results of the proposed model. The comparison results are presented in Figure 5.5. The classifiers are plotted on X-axis and parameters values are plotted on Y-axis. We can observe that the performance of the proposed model is relatively good as compared to the results of all other multi-label classifiers. We have also evaluated different state-of-the-art

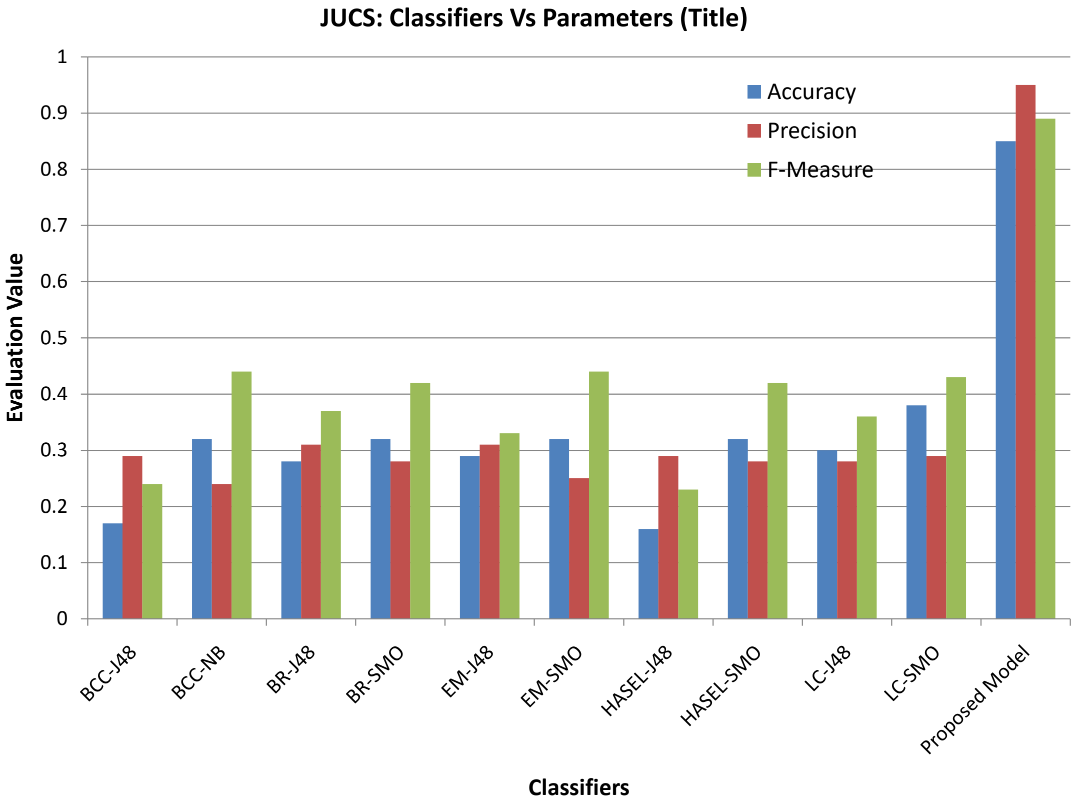


Figure 5.5: J.UCS: Multi-Label Classifiers Vs Evaluation Parameters (*Title*)

multi-label classifiers for metadata (*Title* & *Keywords*) of all research papers in J.UCS dataset; default settings are applied for all multi-label classifiers. Results of all these classifiers are presented in the Figure 5.6; we have compared these multi-label classifiers results with the results of proposed model by plotting these classifiers on X-axis and parameters values on the Y-axis. We can examine that the proposed model outperformed the results of all other multi-label classifiers. The experiments are also performed by using state-of-the-art multi-label classifiers in combination with Naïve Bayes as based classifier on the Topic's References (TR) pairs of J.UCS dataset. For these experiments, there are 10,620 total numbers of instances from which 90% of these are used for training and the remaining ones are used for testing phase, default settings are applied for all state-of-the-art multi-label classifiers and computed the evaluation parameters values. The state-

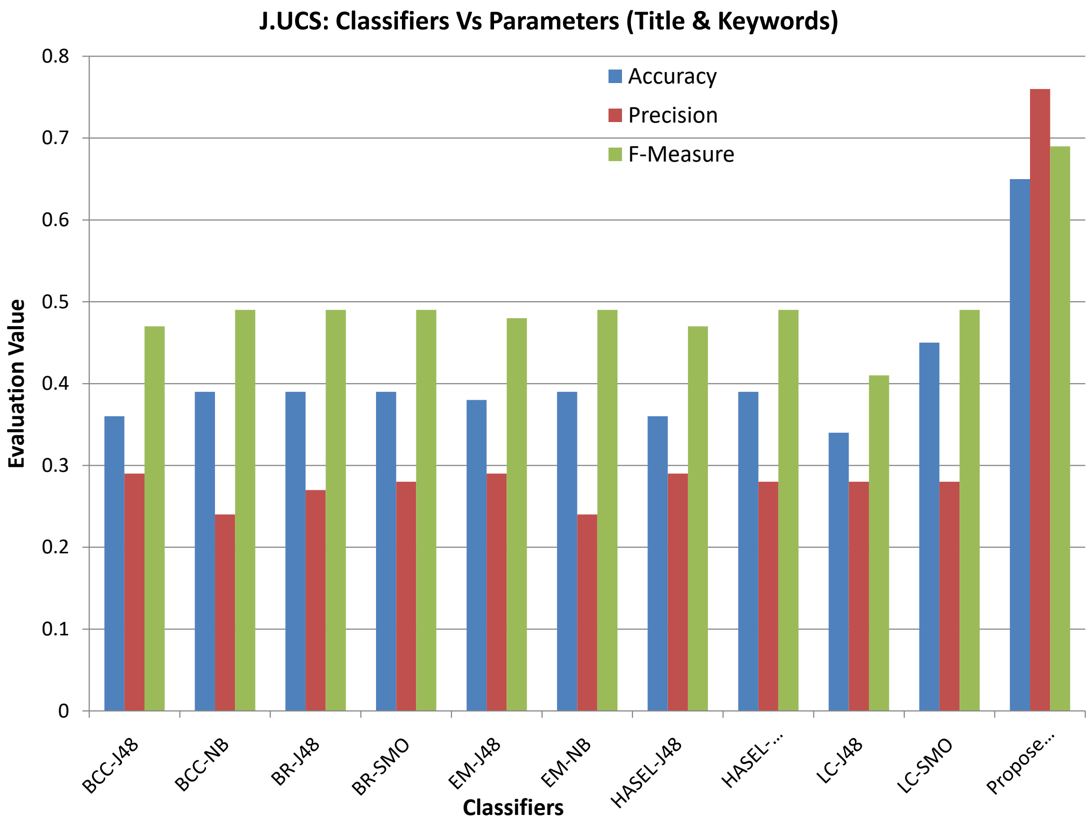


FIGURE 5.6: J.UCS: Multi-Label Classifiers Vs Evaluation Parameters (*Title & Keywords*)

of-the-art multi-label classifiers results are evaluated against the proposed model results and presented in Figure 5.7. The classifier names are plotted on X-axis and evaluation parameter's value on Y-axis. We have critically analyzed the results of state-of-the-art multi-label classifiers for metadata (*Reference's Section*) and concluded the following observations:

- The proposed model outperforms all of the state-of-the-art classifiers results for every evaluation parameter values.
- The performance of all others state-of-the-art multi-label classifiers are consistent on J.UCS dataset. The detailed multi-label classifiers' results for metadata (*Reference's Section*) on J.UCS dataset are shown in Figure 5.7.

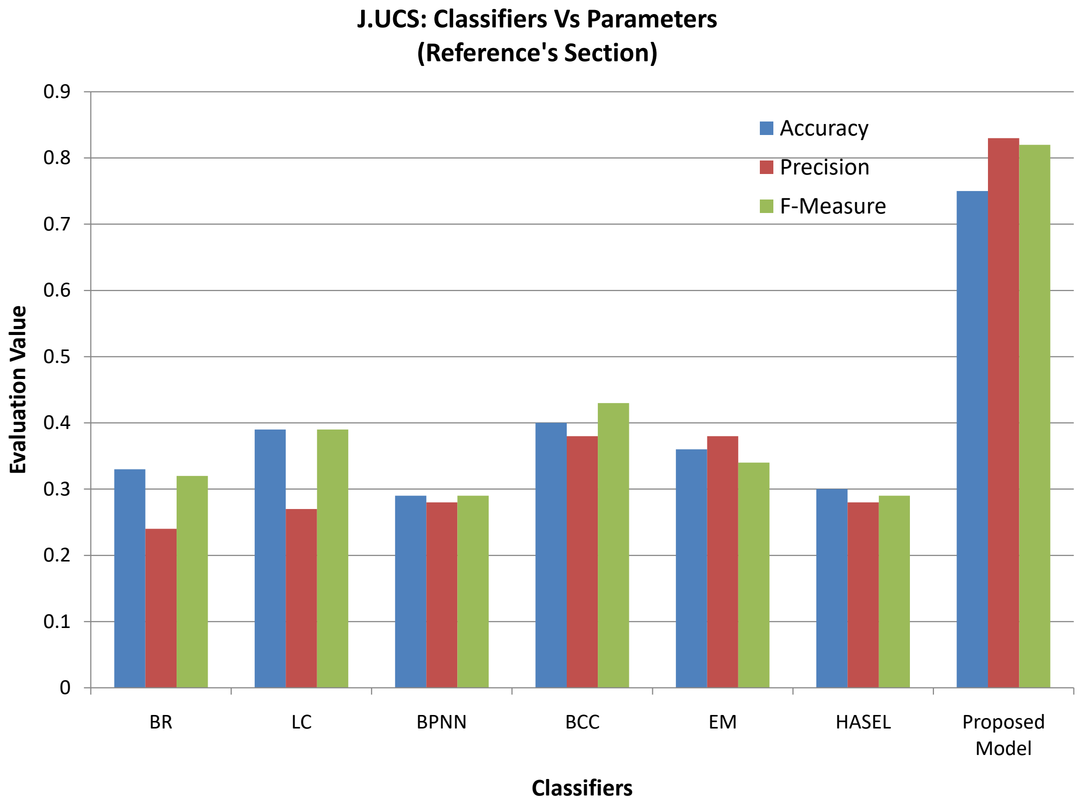


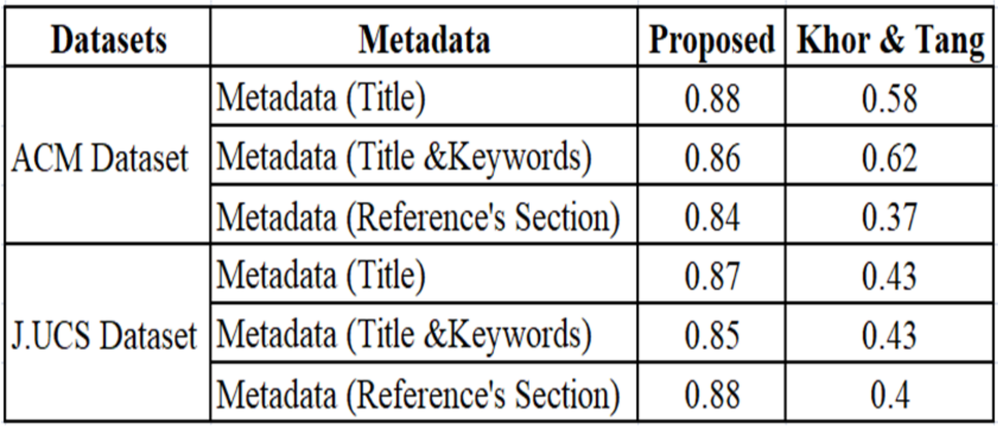
Figure 5.7: J.UCS: Classifiers Vs Evaluation Parameters (*Reference's Section*)

5.2 Comparison with Single-label Document Classification Approach

Single-label document classification approach is proposed by Khor and Tang [14] which utilizes the metadata of the scientific documents. In this approach, 400 educational conference's papers are collected and classified into four topics such as "Intelligent Tutoring System", "Cognition", "E-Learning" and Teacher Education. The keywords are extracted from these papers and some pre-processing steps are applied. These papers are classified into four topics. The performance of different classifiers such as Bayesian Network (BN) [77], Naïve Bayes (NB) and Bayesian Network Learner (BNL) [14] is evaluated with their default settings and the accuracy of 0.84, 0.83 and 0.76 respectively is achieved.

As described earlier, we are interested to scrutinize that to what extent a remarkable accuracy can be achieved by using metadata based features instead of content based features. For this purpose, we have exploited metadata such as *Title*, *Keywords* and *References* from the research papers to classify these papers into multiple categories. In the Chapter 3, we have described the proposed approach

Table 5.2: Comparison with Proposed Approaches



Datasets	Metadata	Proposed	Khor & Tang
ACM Dataset	Metadata (Title)	0.88	0.58
	Metadata (Title &Keywords)	0.86	0.62
	Metadata (Reference's Section)	0.84	0.37
J.UCS Dataset	Metadata (Title)	0.87	0.43
	Metadata (Title &Keywords)	0.85	0.43
	Metadata (Reference's Section)	0.88	0.4

which exploited metadata (*Title*), metadata (*Keywords*) and metadata (*Title & Keywords*) and in the Chapter 4, we have described the proposed approach which exploits the metadata (*Reference's Section*) of the research papers. Experimental results have been conducted on two different and diversified datasets such ACM and J.UCS by selecting different number of research papers. The best results achieved for metadata such as *Title*, *Keywords*, *Title & Keywords* and *Reference* on these two datasets for multi-label classification are presented in the Table 5.2 and these results are compared with the approach proposed by Khor and Tang[14] for single-label classification because this scheme also employs the metadata of the educational conferences papers.

In the Table 5.2, it can be observed that approach proposed by Khor and Tang [14] considered very few numbers of papers as well as classes for the single-label classification. However, the proposed approaches considered large numbers of papers as well as classes for the single-label classification for metadata such as *Title*, *Keywords*, *Title & Keywords* and considered a low number of papers for metadata *References* but used large numbers of classes as compared to the approach proposed by the Khor and Tang [14]. The proposed approach achieved good accuracy for metadata (*Title*) for both datasets and also achieved accuracy equal or greater for other metadata like *Keywords*, *Title & Keywords* and *References* than the approach proposed by Khor and Tang[14]. We have also evaluated our proposed approaches with the state-of-the-art approaches which used different types of classifiers such as Naïve Bayes [34, 41, 45], Bayesian Network [14, 44], SVM [43], Naïve Bayes Multi-nominal etc [18, 78]. We have evaluated these classifiers on both datasets J.UCS and ACM. The critical analysis of the state-of-the-art classifiers, their results and comparison with the proposed approach's results for both datasets are presented in the following sections.

5.2.1 Comparisons on ACM Dataset

We have performed our experiments on ACM dataset in different ways to find out the performance of results. The detailed description and analysis of experiments and their results are as given below:

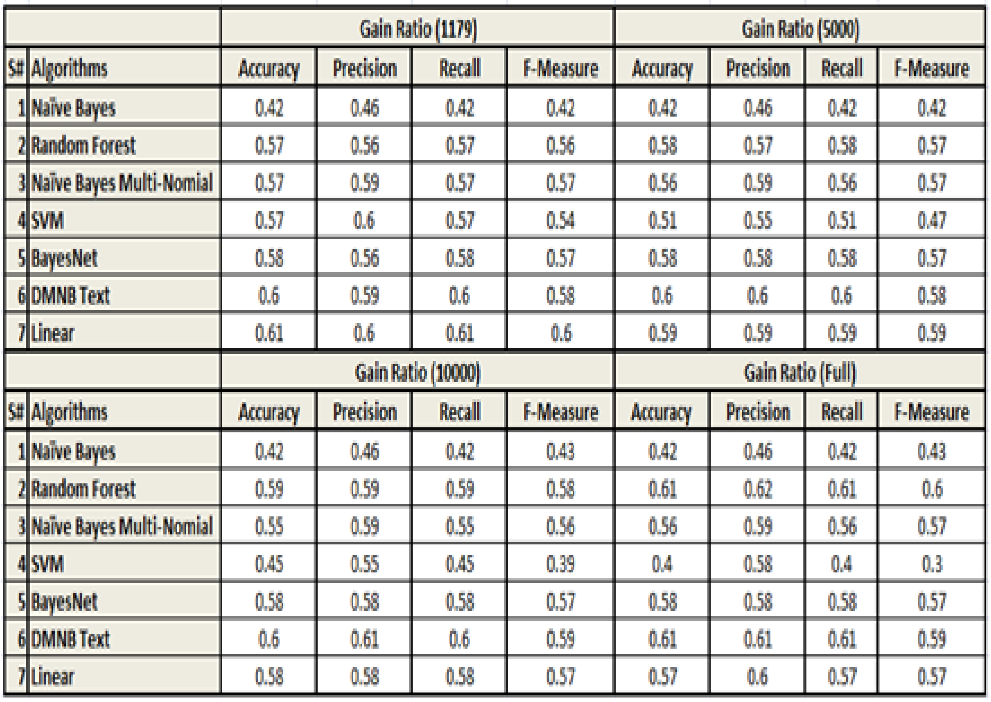
- Experiments on research papers' *Titles*
- Experiments on research papers' *Titles* & *Keywords*
- Experiments on research papers' *References*

5.2.1.1 Experiments on Research Papers' Titles

We have performed experiments on different data sizes of ACM dataset and have evaluated state-of-the-art classifiers. For this purpose, the Gain Ratio is utilized as a feature selection technique to perform the experiments. These features are extracted from the metadata (*Title*) of the research papers. For experimental results, we have performed pre-processing on the dataset and have selected different data sizes. There are 54,994 total number of instances from which 90% instances are chosen for training and 10% instances are chosen for testing phase. The results for the state-of-the-art single-label classifiers are shown in Table 5.3. After the critical analysis of these results, we have concluded the following finding:

- The performance of these classifiers on different features selection is almost consistent.
- Linear classifier performance is better for Gain Ratio (˜1.5K) as compared to other classifiers on evaluation parameters because it performs well on small number of features.
- Discriminative Multinomial Naïve Bayes (DMNB)[79] textual classifier performance is better for Gain Ratio (5K & 10K) as compared to other classifiers on evaluation parameters. DMNB is also good for small number of data sizes.

Table 5.3: ACM Dataset: Results for Metadata (*Title*)



		Gain Ratio (1179)				Gain Ratio (5000)			
S#	Algorithms	Accuracy	Precision	Recall	F-Measure	Accuracy	Precision	Recall	F-Measure
1	Naïve Bayes	0.42	0.46	0.42	0.42	0.42	0.46	0.42	0.42
2	Random Forest	0.57	0.56	0.57	0.56	0.58	0.57	0.58	0.57
3	Naïve Bayes Multi-Nomial	0.57	0.59	0.57	0.57	0.56	0.59	0.56	0.57
4	SVM	0.57	0.6	0.57	0.54	0.51	0.55	0.51	0.47
5	BayesNet	0.58	0.56	0.58	0.57	0.58	0.58	0.58	0.57
6	DMNB Text	0.6	0.59	0.6	0.58	0.6	0.6	0.6	0.58
7	Linear	0.61	0.6	0.61	0.6	0.59	0.59	0.59	0.59
		Gain Ratio (10000)				Gain Ratio (Full)			
S#	Algorithms	Accuracy	Precision	Recall	F-Measure	Accuracy	Precision	Recall	F-Measure
1	Naïve Bayes	0.42	0.46	0.42	0.43	0.42	0.46	0.42	0.43
2	Random Forest	0.59	0.59	0.59	0.58	0.61	0.62	0.61	0.6
3	Naïve Bayes Multi-Nomial	0.55	0.59	0.55	0.56	0.56	0.59	0.56	0.57
4	SVM	0.45	0.55	0.45	0.39	0.4	0.58	0.4	0.3
5	BayesNet	0.58	0.58	0.58	0.57	0.58	0.58	0.58	0.57
6	DMNB Text	0.6	0.61	0.6	0.59	0.61	0.61	0.61	0.59
7	Linear	0.58	0.58	0.58	0.57	0.57	0.6	0.57	0.57

- Similarly, Random Forest classifier performs well among all other classifiers for Gain Ratio (Full).
- In Table 5.3, we can examine that the maximum accuracy, precision, recall and F-measure values achieved by the state-of-the-art classifiers are 0.61, 0.62, 0.61and 0.6 respectively by Random Forest[80] for full Gain Ratio.
- We can also examine that the performance of the classifiers increases as the data size increases.

For evaluation and comparative analysis of the proposed model with the above results of different state-of-the-art classifiers against different sizes of dataset, we have selected the results of state-of-the-art classifiers against the Gain Ratio (Full). We have compared the evaluation parameters results of proposed model for metadata (*Title*) with the results of state-of-the-art classifiers for Gain Ratio (Full). In the Figure 5.8, we have compared state-of-the-art classifiers' results for Gain Ratio (Full) with the proposed model results against the evaluation parameters. We have plotted state-of-the-art classifiers on X-axis and evaluation parameters values on the Y-axis. It can be examined that the proposed model outperforms in all evaluation parameters against the state-of-the-art classifiers. Similarly, proposed

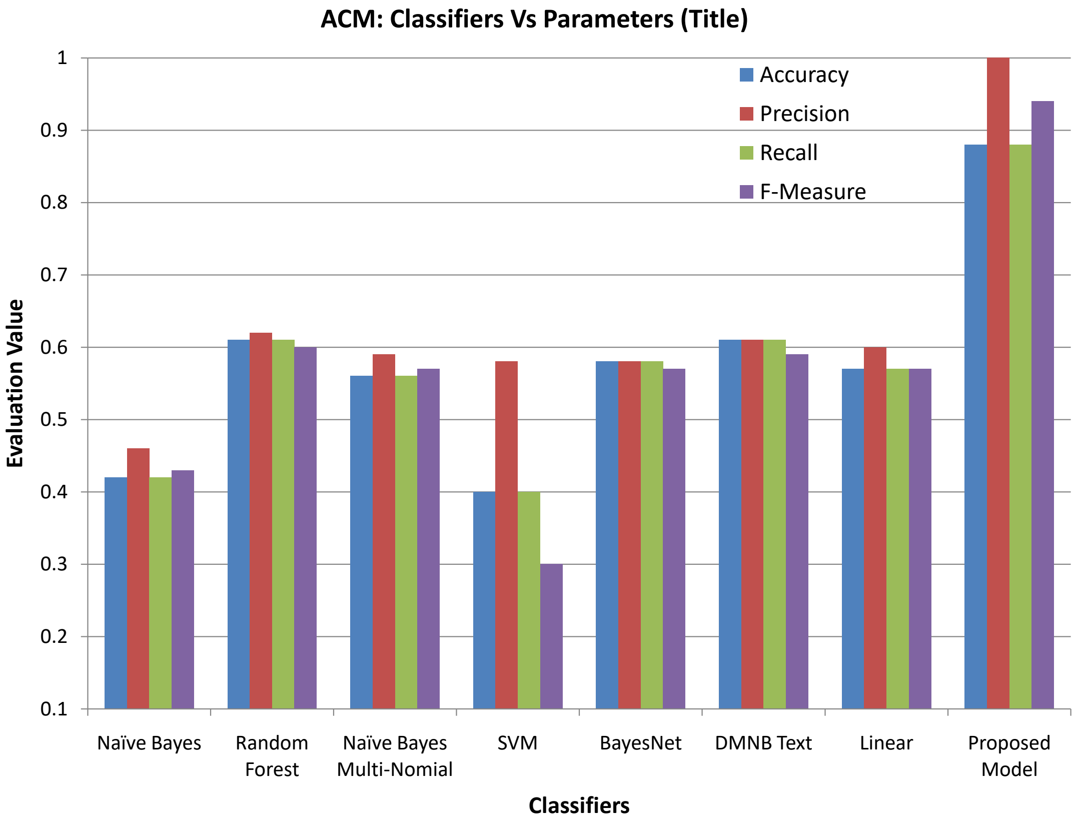


Figure 5.8: ACM: Classifiers Vs Evaluation Parameters (*Title*)

model's results outperform in all evaluation parameters against the state-of-the-art classifiers for other data sizes which are shown above in the Table 5.3. Proposed model's results are good because we have generated a knowledge-base (bags of words) for each category. Test research paper's metadata terms are matched with these bags of words of each category. Proposed model always predicts at least one category which has higher association with the test research paper and the category which has higher number of research papers has more chances to predict for test paper's category. Our knowledge-base is updated regularly with the prediction of categories for the test research papers.

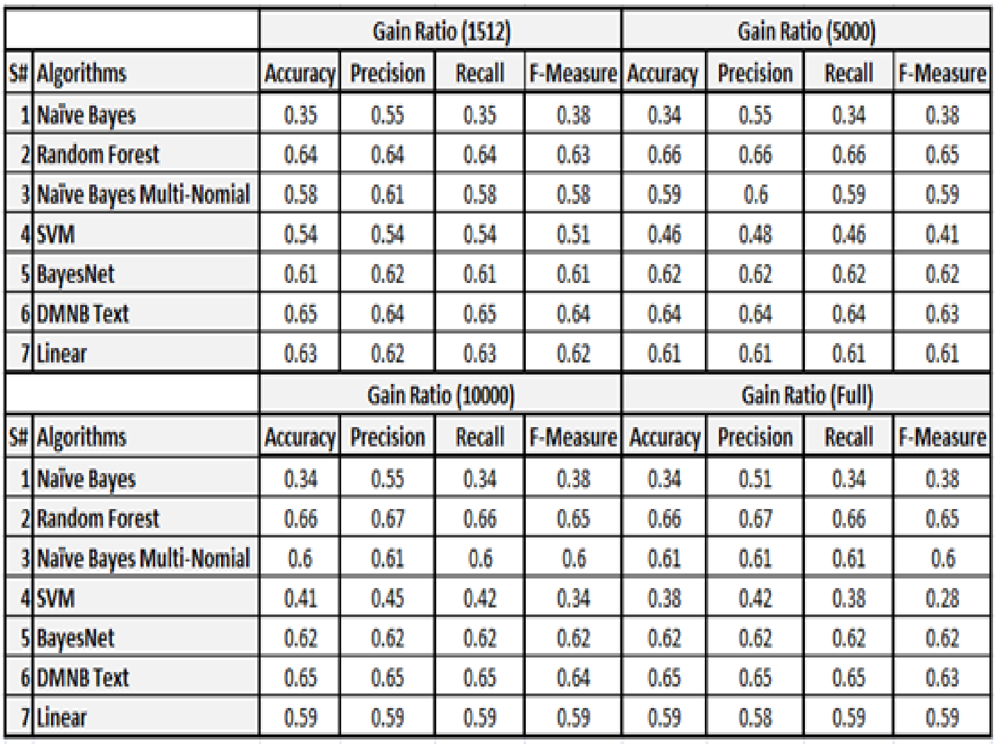
5.2.1.2 Experiments on Research Papers' *Titles & Keywords*

Similar experiments have been performed on different data sizes of ACM dataset and have evaluated state-of-the-art classifiers on different data sizes. We have extracted features from the metadata (*Titles & Keywords*) of the research papers from the ACM dataset. For these experiments, all other configuration settings, parameter selection and data sizes are same as discussed above. The results for the state-of-the-art single-label classifiers are presented in Table 5.4 below. The results are analyzed critically and following points are observed.

- The performance of these classifiers on different features selection is almost consistent.
- Discriminative Multinomial Naïve Bayes (DMNB) text classifier performance is better for Gain Ratio (˜1.5K) as compared to other classifiers on evaluation parameters such as accuracy, precision, recall and F-measure.
- Random Forest classifier performs well among all other classifiers for Gain Ratio (5K, 10K & Full) on all evaluation parameters.
- In Table 5.4, it can be examined that the maximum accuracy, precision, recall and F-measure values achieved by the state-of-the-art classifiers are 0.66, 0.67, 0.66 and 0.65respectively by Random Forest for Gain Ratio (10K).
- The performance of all classifiers is consistent for both Gain Ratios (10K & Full).
- The performance of Naïve Bayes and SVM Classifiers remained low against all parameters for both metadata (*Title*) and metadata (*Titles* & *Keywords*)

For evaluation and comparative analysis of proposed model with the above results of different single-label state-of-the-art classifiers against different sizes of dataset, we have selected the results of state-of-the-art classifiers against the Gain Ratio (Full). The evaluation parameters results of proposed model for metadata (*Titles* & *Keywords*) are compared with the results of state-of-the-art classifiers for GR (Full). The state-of-the-art classifiers' results for Gain Ratio (Full) are compared with the proposed model results against the evaluation parameters. The comparison results are presented in Figure 5.9. The state-of-the-art classifiers are plotted on X-axis and evaluation parameters values are plotted on Y-axis. The proposed model outperforms in all evaluation parameters against the state-of-the-art classifiers. Similarly, proposed model results outperform in all evaluation parameters against the state-of-the-art classifiers for other data sizes which are shown above in Table 5.4.

TABLE 5.4: ACM Dataset: Results for Metadata (*Titles* & *Keywords*)



		Gain Ratio (1512)				Gain Ratio (5000)			
S#	Algorithms	Accuracy	Precision	Recall	F-Measure	Accuracy	Precision	Recall	F-Measure
1	Naïve Bayes	0.35	0.55	0.35	0.38	0.34	0.55	0.34	0.38
2	Random Forest	0.64	0.64	0.64	0.63	0.66	0.66	0.66	0.65
3	Naïve Bayes Multi-Nomial	0.58	0.61	0.58	0.58	0.59	0.6	0.59	0.59
4	SVM	0.54	0.54	0.54	0.51	0.46	0.48	0.46	0.41
5	BayesNet	0.61	0.62	0.61	0.61	0.62	0.62	0.62	0.62
6	DMNB Text	0.65	0.64	0.65	0.64	0.64	0.64	0.64	0.63
7	Linear	0.63	0.62	0.63	0.62	0.61	0.61	0.61	0.61
		Gain Ratio (10000)				Gain Ratio (Full)			
S#	Algorithms	Accuracy	Precision	Recall	F-Measure	Accuracy	Precision	Recall	F-Measure
1	Naïve Bayes	0.34	0.55	0.34	0.38	0.34	0.51	0.34	0.38
2	Random Forest	0.66	0.67	0.66	0.65	0.66	0.67	0.66	0.65
3	Naïve Bayes Multi-Nomial	0.6	0.61	0.6	0.6	0.61	0.61	0.61	0.6
4	SVM	0.41	0.45	0.42	0.34	0.38	0.42	0.38	0.28
5	BayesNet	0.62	0.62	0.62	0.62	0.62	0.62	0.62	0.62
6	DMNB Text	0.65	0.65	0.65	0.64	0.65	0.65	0.65	0.63
7	Linear	0.59	0.59	0.59	0.59	0.59	0.58	0.59	0.59

5.2.1.3 Experiments on Research Papers' References

We have also performed our experiments by exploiting metadata (*Reference's Section*) of the research papers from the ACM dataset and also by applying the above mentioned classifiers. For these experiments, there are 10,065 total numbers of instances from which 90% are used for training and the remaining ones for testing. The default settings are applied for all state-of-the-art single-label classifier and the evaluation parameters values such as accuracy, precision, recall and F-measure are calculated. The detailed results of state-of-the-art single-label classifiers for metadata (*Reference's Section*) on ACM dataset are shown in Figure 5.10. These state-of-the-art classifiers' results are evaluated against the proposed model results. In Figure 5.10, the classifier name is plotted on X-axis and evaluation parameters values are plotted on Y-axis. After the critical analysis of the state-of-the-art single-label classifiers' results on ACM dataset for the metadata (*Reference's Section*), the following points are observed.

- The performance of the proposed model is comparatively good as compared to the state-of-the-art single-label classifiers' results.

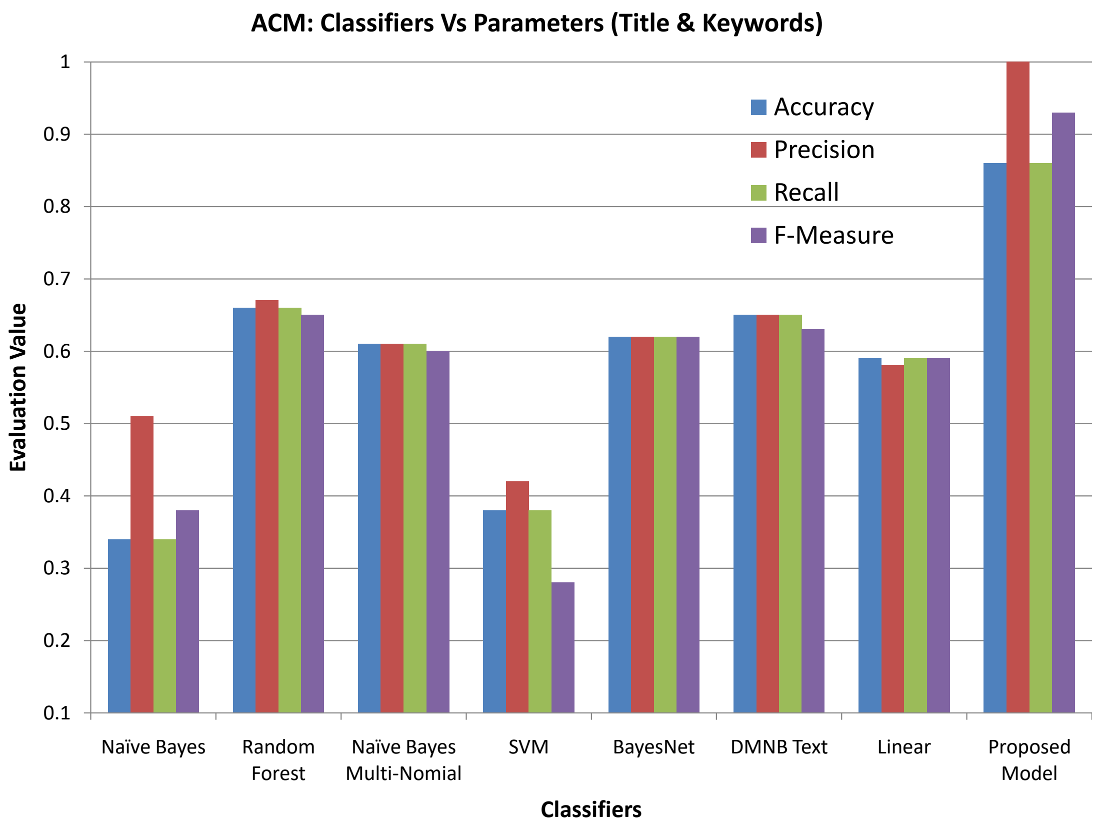


FIGURE 5.9: ACM: Classifiers Vs Evaluation Parameters (*Titles & Keywords*)

- Naïve Bayes Multi-Nomial Text classifier's results are best among all other state-of-the-art classifiers because its performance is significant for textual data.
- All the state-of-the-art single-label classifiers' performance is consistent except Naïve Bayes Multi-Nomial Text Classifier.

5.2.2 Comparison on J.UCS Dataset

Similarly, for the evaluation of proposed technique, the experiments are performed on J.UCS dataset in different ways to find out the performance of proposed model's results. The detailed description and analysis of experiments and their results are given below:

- Experiments on research papers' *Titles*
- Experiments on research papers' *Titles & Keywords*
- Experiments on research papers' *References*

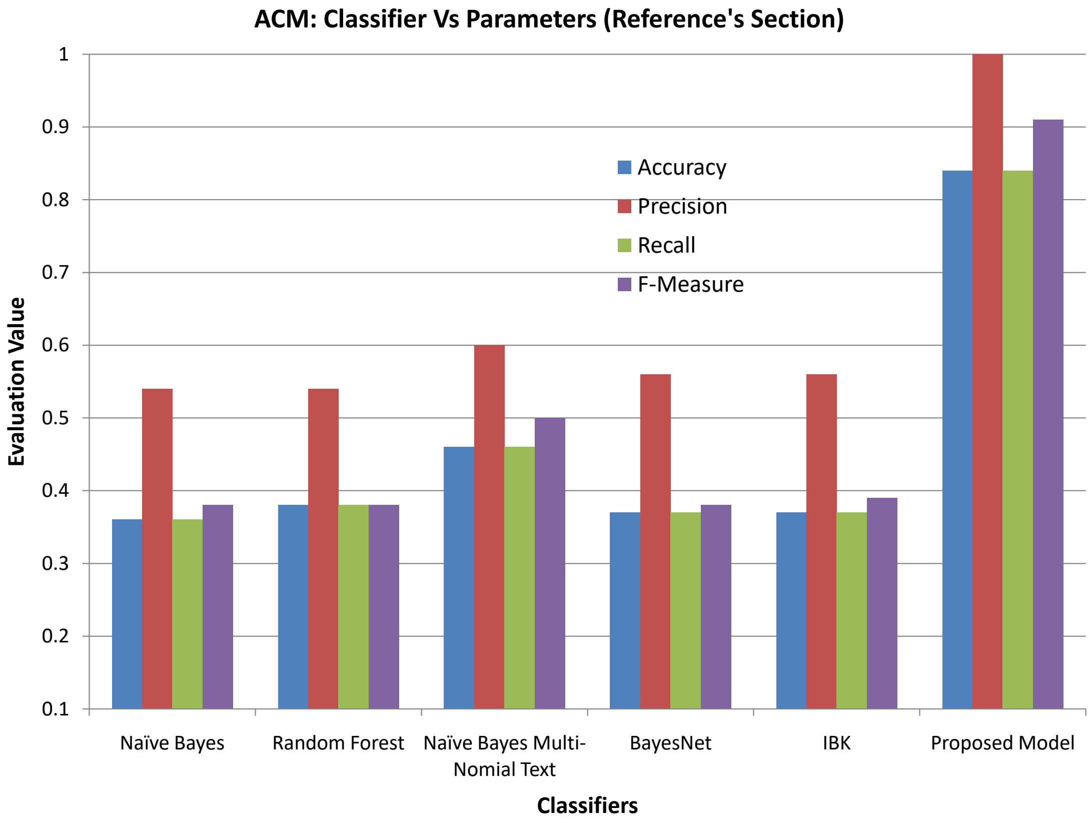


FIGURE 5.10: ACM: Classifiers Vs Evaluation Parameters (*Reference's Section*)

5.2.2.1 Experiments on Research Papers' Titles

We have performed experiments on J.UCS dataset and have evaluated single-label state-of-the-art classifiers on this dataset. Due to a limited number of instances (3,044) in this dataset, full size of J.UCS dataset is taken to perform the experiments. For the evaluation of different single-label state-of-the-art classifiers, two attributes are selected from this dataset which are Titles and Categories of research papers. For the evaluation of experimental results, well-known evaluation parameters such as accuracy, precision, recall and F-measure are chosen. After removing of stop words, each classifier is applied with its default configuration setting and test mode of 90% is used as training set and remaining for testing. Results of single-label state-of-the-art classifiers and proposed model for metadata (*Title*) on J.UCS dataset are presented in the Figure 5.11. In the Figure 5.11, state-of-the-art classifiers are plotted on X-axis and evaluation parameters values are plotted on Y-axis. The evaluation parameters values are computed for each classifier by selecting the metadata (*Title*) of the research papers, default configuration settings are applied for all classifiers. After the critical analysis of the results of single-label state-of-the-art classifiers on J.UCS dataset for the metadata (*Title*), the following points are observed.

- The performance of the various classifiers is almost consistent on the metadata (*Title*) except Naïve Bayes Multi-Nomial classifier which has very low accuracy.
- The proposed model outperforms all of the state-of-the-art Classifiers by achieving significant values against all the evaluation parameters.

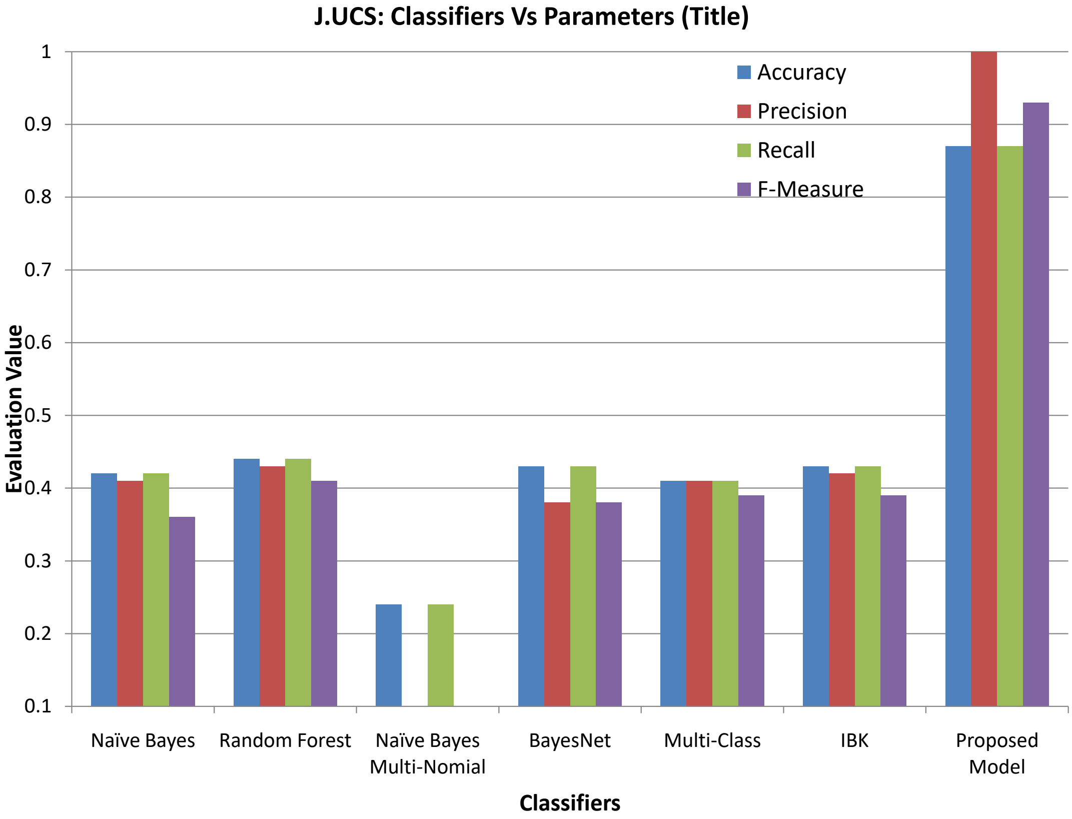


FIGURE 5.11: J.UCS: Classifiers Vs Evaluation Parameters (*Title*)

5.2.2.2 Experiments on Research Papers' *Titles & Keywords*

Similarly, three attributes *Titles*, *Keywords* and *Categories* of these research papers are selected for the evaluation of different single-label state-of-the-art classifiers. The *Title* and *Keywords* strings are merged to form a new string Title_Keywords. For the evaluation of experimental results, same previously utilized evaluation parameters, default configuration settings, removed stop words, test mode of 90% for training set and remaining for testing are utilized for these experiments against metadata (*Titles* & *Keywords*). Results of single-label state-of-the-art classifiers and proposed model for metadata (*Titles* & *Keywords*) on J.UCS dataset are presented in the Figure 5.12. In the Figure 5.12, the single-label state-of-the-art classifiers are plotted on X-axis and evaluation parameters values are plotted on Y-axis. The evaluation parameters values are computed for each classifier by selecting the metadata (*Titles* & *Keywords*) of research papers and applied default configuration setting for all classifiers. After the critical analysis of the state-of-the-art classifiers on J.UCS dataset for the metadata (*Titles* & *Keywords*), we have concluded that:

- The performance of the most classifiers is consistent on the metadata (*Titles* & *Keywords*) except Naïve Bayes Multi-Nomial classifier which has very low accuracy value.
- The proposed model outperforms all of the single-label state-of-the-art classifiers by achieving significant values against all evaluation parameters.

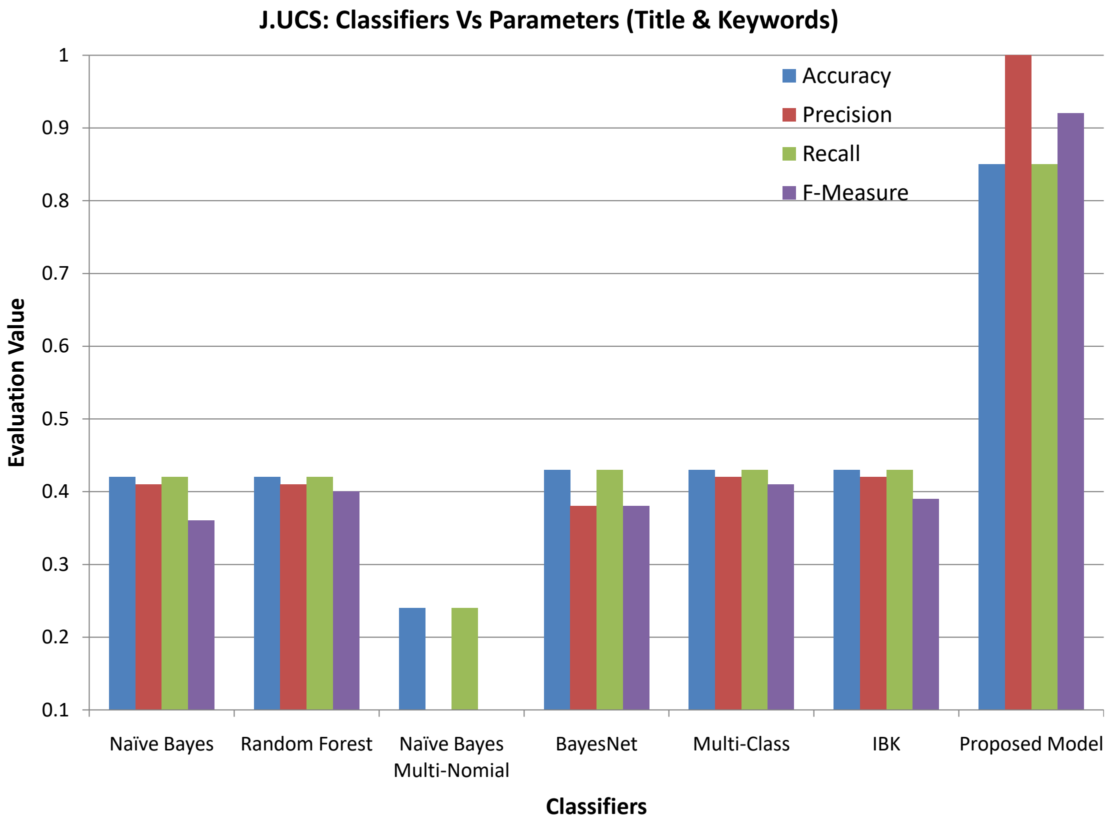


FIGURE 5.12: J.UCS: Classifiers Vs Evaluation Parameters (*Titles* & *Keywords*)

5.2.2.3 Experiments on Research Papers' *References*

Similarly, as described above, the experiments are performed by using state-of-the-art single-label classifiers on the Topic's References (TR) pairs of J.UCS dataset. For these experiments, there are 10,065 total numbers of instances from which 90% are used for training and the remaining ones for testing phase. The default settings are applied for all state-of-the-art classifiers and evaluation parameters values are computed. The state-of-the-art classifiers results are evaluated against the proposed model results which are presented in Figure 5.13. The classifier name is plotted on X-axis and evaluation parameters values are plotted Y-axis. After the critical analysis of the state-of-the-art single-label classifiers' results on J.UCS dataset for the metadata (*Reference's Section*), we have concluded the following points.

- The performance of the proposed model is comparatively good as compared to the state-of-the-art single-label classifiers' results.
- All the state-of-the-art single-label classifiers' performance is consistent except Naïve Bayes Multi-Nomial Text Classifier.

The detailed results of state-of-the-art classifiers for metadata (Reference's Section) on J.UCS dataset are shown in Figure 5.13.

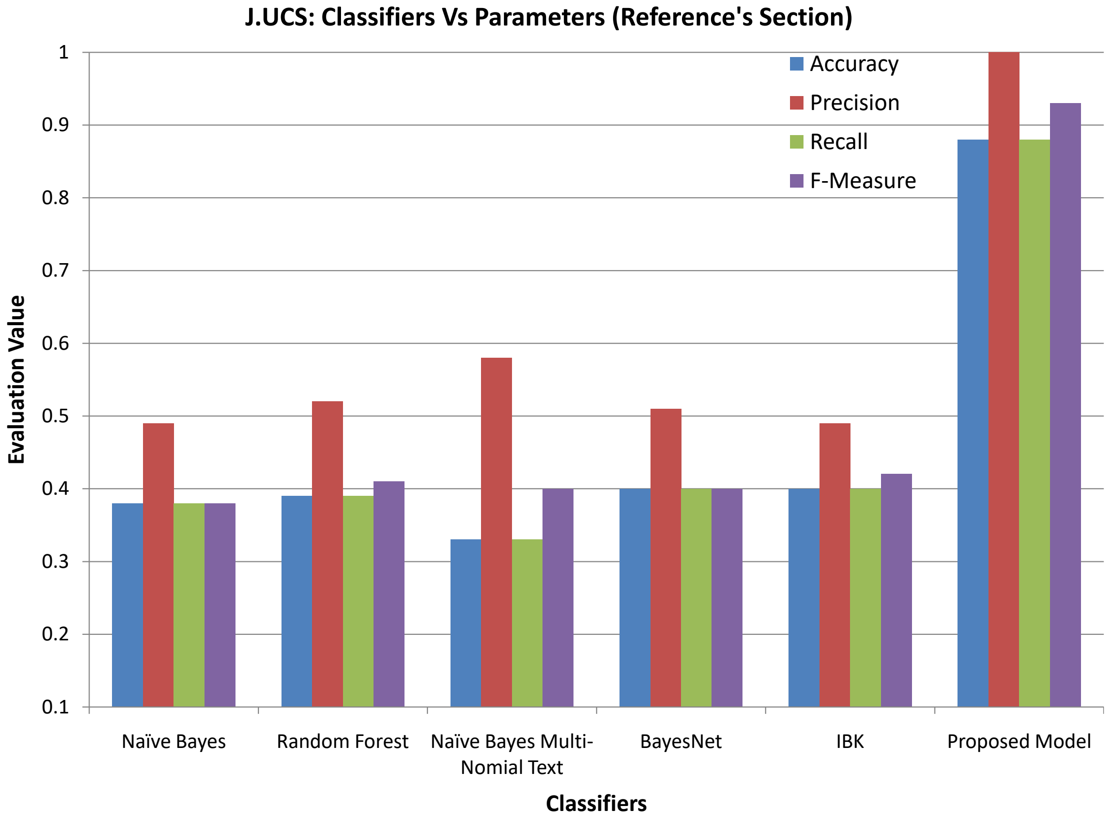


FIGURE 5.13: J.UCS: Classifiers Vs Evaluation Parameters (*Reference's Section*)

Table 5.5: Comparison with State-of-the-art Approaches

S.No.	Approaches	Type of Data	Classification Type	No. of Classes	Dataset	Algorithm / Methodolgy	Evaluation Parameters	Results
1	Galke et al., 2017	Content	Single-Label	Econ (4), Polite (5), RCV1(14), NVT (2)	Econ (62,924), Polite (27,576), RCV1(100,000), NVT (100,000)	KNN	F-Measure	Econ (0.41), Polite (0.27), RCV1(0.76), NVT (0.40)
2	Tang et al., 2016a	Content	Single-Label	20-Newsgroup (20), Reuters (135)	20-Newsgroup (20,000), Reuters (21,578)	Naive Bayes	Accuracy, F-Measure	Accuracy (0.095), F-Measure (0.90)
3	Tang et al., 2016b	Content	Single-Label	20-Newsgroup (20), Reuters (135)	20-Newsgroup (20,000), Reuters (21,578)	Bayesian	F-Measure, G-Mean	Not Reported
4	Shedbale et al., 2016	Content	Single-Label	C, Reuters (135)	20-Newsgroup (20,000), Reuters (21,578)	Survey	Accuracy, F-Measure	Accuracy (0.095), F-Measure (0.90)
5	Zhou, 2016	Content	Single-Label	Not Reported	CiteSeerX (665,483), arXiv (84,172)	Naive Bayes, Logistic Regression	F-Measure	CiteSeerX (0.76), arXiv (0.95)
6	Zong et al., 2015	Content	Single-Label	20-Newsgroup (20), Reuters-10 (10)	20-Newsgroup (16,391), Reuters-10 (7,224)	SVM	F-Measure	20-Newsgroup (0.76), Reuters-10 (0.91)
7	Yaguinuma et al., 2014	Content	Single-Label	4	100 Documents	Fuzz-Onto	Accuracy	Accuracy (0.44)

S.No.	Approaches	Type of Data	Classification Type	No. of Classes	Dataset	Algorithm / Methodolgy	Evaluation Parameters	Results
8	Arash and Mahdi, 2013	Content	Single-Label	wiki-20 (5)	wiki-20 (20)	Concept Matching Based Approach (CMA)	Precision, Recall, F-Measure	Precision (0.61), Recall (0.58), F-Measure (0.60)
9	Hingmire et al, 2013	Content	Single-Label	20-Newsgroup (8), SRAA(10)	20-Newsgroup, SRAA(73,218)	Latent Dirichlet Allocation (LDA)	F-Measure	20-Newsgroup (0.92), SRAA(0.85)
10	Duwairi and Al-Zubaidi, 2011	Content	Single-Label	Not Reported	100 Features	KNN	Precision, Recall	Precision (0.73), Recall (0.55)
11	Wang and Sun, 2009	Content	Single-Label	Reuter (10), WebKB (7)	Reuter (21,578), WebKB (8,282)	NPE, Particle Swarm Optimization (PSO)	F-Measure	Reuter (0.94), WebKB (0.89)
12	Senthamarai and Ramaraj, 2008	Content	Single-Label	Not Reported	2,000 Documents	Particle Swarm Optimization (PSO)	Accuracy	Accuracy (0.90)
13	Jingbo and Tianshun, 2002	Content	Single-Label	10	1000 Documents	Features Identification and Features Aggregation (FIFA)	Precision, Recall	Precision (0.80), Recall (0.76)
14	Flynn, 2014	Metadata	Single-Label	99	2000 Documents	Independent Document Model (IDM) Framework	Precision, Recall	Precision (0.79), Recall (0.81)
15	Zhang et al., 2004	Metadata	Single-Label	11	30,000 Features	Genetic Programming (GP)	Accuracy	Accuracy (0.61)

S.No.	Approaches	Type of Data	Classification Type	No. of Classes	Dataset	Algorithm / Methodolgy	Evaluation Parameters	Results
16	Khor and Ting, 2006	Metadata	Single-Label	4	400 Documents	Baysian Network (BN), Naive Bayes (NB), Baysian Network Learner (BNL)	Accuracy	Accuracy (BN,0.84; NB, 0.83; BNL, 0.76)
17	Yan et al., 2018	Content	Multi-Label	Biomedicine (150), Email (6), News(103)	Biomedicine (100,000), Email (3, 021), News(800,000)	Long Short Term Memory (LSTM)	F-Measure	F-Measure (0.70)
18	Wang et al., 2018	Content	Multi-Label	Yahoo (30)	Yahoo Features (700)	CLSVCM	Accuracy	Accuracy (0.85)
19	Baker and Korhonen, 2017	Content	Multi-Label	PubMed (30)	PubMed (1,852)	INIT-A, INIT-B	Precision, Recall, F-Measure	Precision (0.73, 0.68), Recall (0.77, 0.83), F-Measure (0.75, 0.75)
20	Santos and Rodrigues., 2009	Content	Multi-Label	11	5,000 and 10,000 Documents	Binary Relevence (BR), Label Power (LP), Multi-Label kNN (MLkNN)	Accuracy	Accuracy (0.88)
21	Lijuan, 2008	Content	Multi-Label	WIPO-alpha (8), Newsgroup (5), OHSUMED (15), ENZYME (236)	Synthetic Data, WIPO-alpha, Newsgroup (1000),OHSUMED (54,708), ENZYME (9,455)	Hierarchical SVM, Hierarchical Perception	Accuracy, Precision	Accuracy (0.94), Precision (0.89)

S.No.	Approaches	Type of Data	Classification Type	No. of Classes	Dataset	Algorithm / Methodolgy	Evaluation Parameters	Results
22	Wang and Desai, 2007	Content	Multi-Label	6	45,000 Features	Naive Bayes, Centroid	Accuracy	Accuracy (0.61)
23	Proposed Approach (1)	Metadata	Multi-Label	11	JUCS (1,460), ACM (86,116)	Algorithm Based on Term Frequency (TF)	Accuracy, Precision, Recall, F-Measure	Accuracy (0.88), Precision (0.91), Recall (0.94), F-Measure (0.92)
24	Proposed Approach (2)	Metadata	Multi-Label	11	JUCS (1,460), ACM (86,116)	Algorithm Based on Topic's Reference (TR) Pair	Accuracy, Precision, Recall, F-Measure	Accuracy (0.74), Precision (0.78), Recall (0.81), F-Measure (0.79)

5.3 Summary

In this chapter, we have critically evaluated and compared the results of proposed approaches which exploit only metadata of the research papers to perform single-label as well as multi-label document classification. Comparison with the content based multi-label approach depicted that the proposed multi-label approaches perform significantly closer or equal in terms of accuracy by relying only on metadata based features. Similarly, the proposed approaches for single-label document classification outperform the state-of-the-art single-label classification approaches. Moreover, the proposed approaches are evaluated with the state-of-the-art single-label as well as multi-label approaches which utilize different types of classifier and concluded that proposed approaches also outperform the state-of-the-art classifiers' results on two different and diversified datasets. Comparison with the state-of-the-art approaches is presented in Table 5.5.

Chapter 6

Conclusion and Future Work

6.1 Conclusion

Classification of research articles into pre-defined categories is a daunting research challenge and has many applications such as: selection of reviewers; indexing and retrieval of relevant research papers; paper submission; and expertise discovery applications. There exist enormous approaches in the literature to categorize research papers. The critical analysis of the literature has disclosed that majority of such systems utilize content of the papers. However, the un-availability of content renders these schemes non-applicable most of the time. Moreover, the contemporary approaches classify papers into single class. We argue that a research paper may belong to various categories due to the diversified linkage between multiple disciplines. These issues have led us to scrutinize the potential of freely available metadata to discover efficient adaptable ways in the scenarios when the content is not available.

This dissertation is aimed to utilize freely available metadata and evaluate the same so as to discover that to what extent metadata can be used to classify research papers into multi-label classification by employing the comprehensive novel mechanisms. In this dissertation, we have proposed, developed and evaluated approaches on metadata based features to overcome the above mentioned issues. For classification of research articles based on metadata and into multi-labels, we had exploited metadata in different ways: (1) Multi-label Document Classification on Papers' Metadata (*Title* & *Keywords*) and (2) Multi-label Document Classification based on papers' *References*. These approaches have been evaluated against two different diversified datasets such as J.UCS [37] and ACM [18]. Subsequently, the proposed approaches were compared with state-of-the-art approaches and classifiers.

Chapter 3 of this dissertation addressed a novel approach "*Multi-label Document Classification on Papers' Metadata (Title & Keywords)*" [36] which extracted metadata *Title* and *Keywords* from the research papers and Multi-label Classifier (MLC) performed multi-label classification on both datasets in three ways:

- By extracting only metadata (*Titles*) from the research papers.
- By extracting only metadata (*Keywords*) from the research papers.
- By extracting both metadata (*Title* & *Keywords*) of the research papers.

For each research paper's metadata *Title*, *Keywords* and both *Title* & *Keywords* the term frequency weights are computed similarly, after generating the research paper's category pairs, the term frequency weights are computed for each category provided by the datasets for the same metadata. The experimental results of this approach are encouraging because in the said approach, we have utilized only freely available metadata. The best threshold value and the best metadata parameter, at which the proposed approach produces significant results, are identified by conducting multiple experimental rounds. At the best threshold value, the approach has achieved an accuracy of 0.88 by exploiting only metadata (*Title*) which is exactly same as achieved by Santos and Rodrigues [18] but they have utilized the whole content of the scientific documents. Similarly, other metadata such metadata (*Keywords*) has achieved accuracy of 0.73, metadata (*Title* & *Keywords*) has achieved accuracy of 0.79 and metadata (*References*) has achieved accuracy of 0.71 which are also very close to the results of Santos and Rodrigues [18]. Hence, the proposed system has achieved good accuracy by exploiting only metadata of the scientific documents. Single-label classification is also performed by exploiting above mentioned metadata parameters and system has achieved accuracy 0.88 which is quite remarkable as compared to the state-of-the-art metadata based classification approaches which has achieved maximum accuracy of 0.84 by Khor and Tang [14].

Chapter 4 illustrated the second approach "*Multi-label Document Classification based on Paper's References*" [10]. In this approach, the research paper's references are extracted from the web links provided in both datasets. The topic's reference pairs are generated for comparison of the test paper's references. The Citation Based Category Identification (CBCI) classifier compares the test paper's references with the stored category-wise references and performs multi-label classification. The experimental results of this approach also yielded encouraging results by exploiting freely available metadata (*Reference's Section*) as compared to the state-of-the-art approaches for multi-label as well as for single-label document classification. This system has achieved accuracy of 0.74 for multi-label classification and accuracy of 0.88 for single-label classification which are quite unprecedented if we consider the fact that the value is achieved by using only metadata based features.

6.2 Dissertation's Contributions

For research article classification, this dissertation completely relies on freely available metadata of the documents instead of using the whole content of the documents. The contributions of this dissertation are following:

1. A comprehensive survey is conducted covering manifold state-of-the-art approaches which performed single-label or multi-label classification by exploiting either metadata or content of the research articles.

2. State-of-the-art classification approaches are critically analyzed and evaluated against evaluation criteria inferred from the literature review.

3. A novel approach is proposed, implemented and evaluated by exploiting only metadata (*Title* and *Keywords*) instead of the whole content of the research papers.

4. Another novel approach is proposed, implemented and evaluated by exploiting only metadata (*Reference's Section*) of the research papers.

5. The proposed approaches are able to classify papers into both a single-label and multi-label classes by obtaining an immense place in current state-of-the-art document classification community.

6. The proposed approaches are evaluated on comprehensive datasets (J.UCS [37] & ACM [18]) containing research articles belonging to one or more than one category.

7. The best threshold value is recommended for the multi-label document classification and the best metadata parameter is recommended for single-label as well as for multi-label document classification.

8. Evaluation and comparison of the proposed approaches are conducted against state-of-the-art classification approaches.

6.3 Future Work

Some potential directions for future research on the area addressed in this dissertation are described below:

- This work can be extended in the future by exploiting metadata information to perform multi-label classification for next ACM taxonomy levels.
- Evolutionary approaches can be used to exploit metadata for multi-label document classification.
- Evaluation of proposed approaches in domains other than Computer Science

Bibliography

[1] L. Bornmann and R. Mutz, "Growth rates of modern science: A bibliometric analysis based on the number of publications and cited references," *Journal of the Association for Information Science and Technology*, vol. 66, no. 11, pp. 2215–2222, 2015.

[2] P. O. Larsen and M. Von Ins, "The rate of growth in scientific publication and the decline in coverage provided by science citation index," *Scientometrics*, vol. 84, no. 3, pp. 575–603, 2010.

[3] J. Davies, R. Weeks, and M. Revett, "Jasper: Communicating information agents," in *Proceedings of the 4th International Conference on the World Wide Web*, 1995.

[4] A. Hodgson and L. Schlager, "Closing the pdf gap: Readcube's experiments in reader-focused design," vol. 30, no. 1. Wiley Online Library, 2017, pp. 65–69.

[5] M. Ware and M. Mabe, "The stm report: An overview of scientific and scholarly journal publishing," *Oxford, UK: International Association of Scientific, Technical and Medical Publishers*, 2015.

[6] D. Koller and M. Sahami, "Hierarchically classifying documents using very few words," in *Stanford InfoLab*, 1997, pp. 170–178.

[7] F. Sebastiani, "Machine learning in automated text categorization," *ACM computing surveys (CSUR)*, vol. 34, no. 1, pp. 1–47, 2002.

[8] I. Kononenko, "Comparison of inductive and naive bayesian learning approaches to automatic knowledge acquisition," *Current trends in knowledge acquisition*, pp. 190–197, 1990.

[9] M. F. Porter, "An algorithm for suffix stripping," in *Readings in Information Retrieval, Morgan Kaufmann Publishers Inc., San Francisco, CA*, 1997, pp. 313–316.

[10] N. A. Sajid, T. Ali, M. T. Afzal, M. Ahmad, and M. A. Qadir, "Exploiting reference section to classify paper's topics," in *Proceedings of the International Conference on Management of Emergent Digital EcoSystems.* ACM, 2011, pp. 220–225.

[11] C. Goller, J. Löning, T. Will, and W. Wolff, "Automatic document classification-a thorough evaluation of various methods." *ISI*, vol. 2000, pp. 145–162, 2000.

[12] V. P. G. Bote, F. de Moya Anegón, and V. H. Solana, "Document organization using kohonen's algorithm," *Information processing & management*, vol. 38, no. 1, pp. 79–89, 2002.

[13] B. Zhang, M. A. Gonçalves, W. Fan, Y. Chen, E. A. Fox, P. Calado, and M. Cristo, "Combining structural and citation-based evidence for text classification," in *Proceedings of the thirteenth ACM international conference on Information and knowledge management.* ACM, 2004, pp. 162–163.

[14] K.-C. Khor and C.-Y. Ting, "A bayesian approach to classify conference papers," in *Mexican International Conference on Artificial Intelligence.* Springer, 2006, pp. 1027–1036.

[15] O. Dehzangi, M. J. Zolghadri, S. Taheri, and S. M. Fakhrahmad, "Efficient fuzzy rule generation: A new approach using data mining principles and rule weighting," in *Fuzzy Systems and Knowledge Discovery, 2007. FSKD 2007. Fourth International Conference on*, vol. 2. IEEE, 2007, pp. 134–139.

[16] T. Li, S. Zhu, and M. Ogihara, "Hierarchical document classification using automatically generated hierarchy," *Journal of Intelligent Information Systems*, vol. 29, no. 2, pp. 211–230, 2007.

[17] S. S. Karman and N. Ramaraj, "Similarity-based techniques for text document classification," *Int. J. SoftComput*, vol. 3, no. 1, pp. 58–62, 2008.

[18] A. P. Santos and F. Rodrigues, "Multi-label hierarchical text classification using the acm taxonomy," in *14th Portuguese Conference on Artificial Intelligence (EPIA)*, 2009, pp. 553–564.

[19] W. Ziqiang and S. Xia, "Document classification algorithm based on npe and pso," *E-Business and Information System Security*, pp. 1–4, 2009.

[20] A. Joorabchi and A. E. Mahdi, "Classification of scientific publications according to library controlled vocabularies: a new concept matching-based approach," *Library Hi Tech*, vol. 31, no. 4, pp. 725–747, 2013.

[21] S. Hingmire, S. Chougule, G. K. Palshikar, and S. Chakraborti, "Document classification by topic labeling," in *Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval.* ACM, 2013, pp. 877–880.

[22] P. J. Dendek, A. Czeczko, M. Fedoryszak, A. Kawa, P. Wendykier, and Ł. Bolikowski, "Content analysis of scientific articles in apache hadoop ecosystem," in *Intelligent Tools for Building a Scientific Information Platform: From Research to Implementation.* Springer, 2014, pp. 157–172.

[23] T. Giannakopoulos, E. Stamatogiannakis, I. Foufoulas, H. Dimitropoulos, N. Manola, and Y. Ioannidis, "Content visualization of scientific corpora using an extensible relational database implementation," in *International Conference on Theory and Practice of Digital Libraries.* Springer, 2014, pp. 101–112.

[24] L. C. Smith, "Citation analysis," *Library Trends*, vol. 30, no. 1, pp. 83–106, 1981.

[25] F. Jelinek, *Statistical methods for speech recognition.* MIT press, 1998.

[26] C. Apté, F. Damerau, and S. M. Weiss, "Automated learning of decision rules for text categorization," *ACM Transactions on Information Systems (TOIS)*, vol. 12, no. 3, pp. 233–251, 1994.

[27] I. Dagan, Y. Karov, and D. Roth, "Mistake-driven learning in text categorization," *arXiv preprint cmp-lg/9706006*, 1997.

[28] K. Shin, A. Abraham, and S.-Y. Han, "Enhanced centroid-based classification technique by filtering outliers," in *TSD*, vol. 6. Springer, 2008, pp. 159–163.

[29] G. Salton, "Developments in automatic text retrieval," *science, American Association for the Advancement of Science*, vol. 253, no. 5023, pp. 974–980, 1991.

[30] P. Gerstl, M. Hertweck, and B. Kuhn, "Text mining: Grundlagen, verfahren und anwendungen," *HMD-Praxis der Wirtschaftsinformatik*, vol. 38, no. 222, pp. 38–48, 2001.

[31] S. Har-Peled, D. Roth, and D. Zimak, "Constraint classification for multiclass classification and ranking," in *Advances in neural information processing systems*, 2002, pp. 809–816.

[32] N. Coulter, J. French, E. Glinert, T. Horton, N. Mead, R. Rada, A. Ralston, C. Rodkin, B. Rous, A. Tucker *et al.*, "Computing classification system 1998: current status and future maintenance. report of the ccs update committee," *Computing Reviews*, vol. 39, no. 1, pp. 1–62, 1998.

[33] B. Shneiderman, D. Feldman, A. Rose, and X. F. Grau, "Visualizing digital library search results with categorical and hierarchical axes," in *Proceedings of the fifth ACM conference on Digital libraries.* ACM, 2000, pp. 57–66.

[34] T. Wang and B. C. Desai, "Document classification with acm subject hierarchy," in *Electrical and Computer Engineering, 2007. CCECE 2007. Canadian Conference on.* IEEE, 2007, pp. 792–795.

[35] P. K. Flynn, *Document classification in support of automated metadata extraction form heterogeneous collections.* Old Dominion University, 2014.

[36] N. Sajid, M. Afzal, and M. Qadir, "Multi-label classification of computer science documents using fuzzy logic," *Journal of the National Science Foundation of Sri Lanka*, vol. 44, no. 2, pp. 155–165, 2016.

[37] M. T. Afzal, N. Kulathuramaiyer, and H. A. Maurer, "Creating links into the future." *J. UCS*, vol. 13, no. 9, pp. 1234–1245, 2007.

[38] N. A. Sajid, M. T. Afzal, M. A. Qadir, and S. A. Khan, "The insights of classification schemes," *Sindh University Research Journal (Science Series)*, vol. 45, no. A-1, pp. 145–150, 2013.

[39] R. Kumar, *Research Methodology: a Step-by-Step Guide for Beginners.* 3rd Edition, SAGE Publication Limited, 2011.

[40] N. Zechner, "The past, present and future of text classification," *methods*, vol. 1, p. 4, 2013.

[41] B. Tang, S. Kay, and H. He, "Toward optimal feature selection in naive bayes for text categorization," *IEEE Transactions on Knowledge and Data Engineering*, vol. 28, no. 9, pp. 2508–2521, 2016.

[42] S. Shedbale, K. Shaw, and P. K. Mallick, "Filter feature selection approaches for automated text categorization," *International Journal of Control Theory and Application*, vol. 10, no. 8, pp. 763–773, 2016.

[43] W. Zong, F. Wu, L.-K. Chu, and D. Sculli, "A discriminative and semantic feature selection method for text categorization," *International Journal of Production Economics*, vol. 165, pp. 215–222, 2015.

[44] B. Tang, H. He, P. M. Baggenstoss, and S. Kay, "A bayesian classification approach using class-specific features for text categorization," *IEEE Transactions on Knowledge and Data Engineering*, vol. 28, no. 6, pp. 1602–1606, 2016.

[45] T. Zhou, "Automated identification of computer science research papers," in *University of Windsor (Canada)*, 2016.

[46] A. R. Afonso and C. G. Duque, "Automated text clustering of newspaper and scientific texts in brazilian portuguese: analysis and comparison of methods," *JISTEM-Journal of Information Systems and Technology Management*, vol. 11, no. 2, pp. 415–436, 2014.

[47] C. A. Yaguinuma, M. T. Santos, H. A. Camargo, M. Nicoletti, and T. M. Nogueira, "A meta-ontology for modeling fuzzy ontologies and its use in classification tasks based on fuzzy rules," *International Journal of Computer Information Systems and Industrial Management Applications*, vol. 6, pp. 89–101, 2014.

[48] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *Journal of machine Learning research*, vol. 3, no. Jan, pp. 993–1022, 2003.

[49] F. M. Ortuño, I. Rojas, M. A. Andrade-Navarro, and J.-F. Fontaine, "Using cited references to improve the retrieval of related biomedical documents," *BMC bioinformatics*, vol. 14, no. 1, p. 113, 2013.

[50] R. M. Duwairi and R. Al-Zubaidi, "A hierarchical k-nn classifier for textual data." *Int. Arab J. Inf. Technol.*, vol. 8, no. 3, pp. 251–259, 2011.

[51] S. Eyheramendy and D. Madigan, “A novel feature selection score for text categorization,” in *Proceedings of the Workshop on Feature Selection for Data Mining, in conjunction with the 2005 SIAM International Conference on Data Mining*, 2005, pp. 1–8.

[52] B. Tang, M. Shepherd, E. Milios, and M. I. Heywood, “Comparing and combining dimension reduction techniques for efficient text clustering,” in *Proceeding of SIAM International Workshop on Feature Selection for Data Mining*, 2005, pp. 17–26.

[53] L. Galke, F. Mai, A. Schelten, D. Brunsch, and A. Scherp, “Using titles vs. full-text as source for automated semantic document annotation,” in *Proceedings of the Knowledge Capture Conference.* ACM, 2017, p. 20.

[54] L. Cai, *Multilabel classification over category taxonomies.* Brown University, 2008.

[55] L. Cai and T. Hofmann, “Hierarchical document categorization with support vector machines,” in *Proceedings of the thirteenth ACM international conference on Information and knowledge management.* ACM, 2004, pp. 78–87.

[56] Y. Yan, Y. Wang, W.-C. Gao, B.-W. Zhang, C. Yang, and X.-C. Yin, “Lstm2: Multi-label ranking for document classification,” *Neural Processing Letters*, vol. 47, no. 1, pp. 117–138, 2018.

[57] R. Wang, G. Chen, and X. Sui, “Multi label text classification method based on co-occurrence latent semantic vector space,” *Procedia Computer Science*, vol. 131, pp. 756–764, 2018.

[58] S. Baker and A. Korhonen, “Initializing neural networks for hierarchical multilabel text classification,” *BioNLP 2017*, pp. 307–315, 2017.

[59] H. Brücher, G. Knolmayer, and M.-A. Mittermayer, “Document classification methods for organizing explicit knowledge,” 2002.

[60] P. Yohan, B. Sasidhar, S. A. H. Basha, and A. Govardhan, “Automatic named entity identification and classification using heuristic based approach for telugu,” *International Journal of Computer Science Issues*, vol. 11, no. 1, 2014.

[61] Z. Jingbo and Y. Tianshun, “A knowledge-based approach to text classification,” in *Proceedings of the first SIGHAN workshop on Chinese language*

processing-Volume 18. Association for Computational Linguistics, 2002, pp. 1–5.

[62] S. Godbole and S. Sarawagi, "Discriminative methods for multi-labeled classification," *Advances in knowledge discovery and data mining*, pp. 22–30, 2004.

[63] E. Garfield *et al.*, "Can citation indexing be automated," in *Statistical association methods for mechanized documentation, symposium proceedings*, vol. 269. National Bureau of Standards, Miscellaneous Publication 269, Washington, DC, 1965, pp. 189–192.

[64] A. Shahid, M. T. Afzal, and M. A. Qadir, "Lessons learned: The complexity of accurate identification of in-text citations." *Int. Arab J. Inf. Technol.*, vol. 12, no. 5, pp. 481–488, 2015.

[65] M. T. Afzal, W.-T. Balke, H. Maurer, and N. Kulathuramaiyer, "Improving citation mining," in *Networked Digital Technologies, 2009. NDT'09. First International Conference on.* IEEE, 2009, pp. 116–121.

[66] V. I. Levenshtein, "Binary codes capable of correcting deletions, insertions, and reversals," in *Soviet physics doklady*, vol. 10, no. 8, 1966, pp. 707–710.

[67] K. Rieck and C. Wressnegger, "Harry: a tool for measuring string similarity," *Journal of Machine Learning Research*, vol. 17, no. 9, pp. 1–5, 2016.

[68] S. P. Singh, A. Kumar, H. Darbari, S. Chauhan, N. Srivastava, and P. Singh, "Evaluation of similarity metrics for translation retrieval in the hindi-english translation memory," *Evaluation*, vol. 4, no. 8, 2015.

[69] C. Lijuan, *Similarity Measures.* Hasso plattner Institute, 2013.

[70] S. M. A. Elrahman and A. Abraham, "A review of class imbalance problem," *Journal of Network and Innovative Computing*, vol. 1, no. 2013, pp. 332–340, 2013.

[71] G. Tsoumakas and I. Katakis, "Multi-label classification: An overview," *International Journal of Data Warehousing and Mining*, vol. 3, no. 3, 2007.

[72] M.-L. Zhang and Z.-H. Zhou, "A k-nearest neighbor based algorithm for multi-label classification," in *Granular Computing, 2005 IEEE International Conference on*, vol. 2. IEEE, 2005, pp. 718–721.

[73] C. B. Do and S. Batzoglou, "What is the expectation maximization algorithm?" *Nature biotechnology*, vol. 26, no. 8, pp. 897–899, 2008.

[74] M. Grobelnik, "Feature selection for unbalanced class distribution and naive bayes." ICML, 1999, pp. 258–267.

[75] K. M. Leung, "Naive bayesian classifier," *Polytechnic University Department of Computer Science/Finance and Risk Engineering*, 2007.

[76] J. Platt, "Sequential minimal optimization: A fast algorithm for training support vector machines," 1998.

[77] J. Pearl, "Morgan kaufmann series in representation and reasoning. probabilistic reasoning in intelligent systems: Networks of plausible inference," 1988.

[78] A. M. Kibriya, E. Frank, B. Pfahringer, and G. Holmes, "Multinomial naive bayes for text categorization revisited." in *Australian Conference on Artificial Intelligence*, vol. 3339. Springer, 2004, pp. 488–499.

[79] M. Panda, A. Abraham, and M. R. Patra, "Discriminative multinomial naive bayes for network intrusion detection," in *Information Assurance and Security (IAS), 2010 Sixth International Conference on.* IEEE, 2010, pp. 5–10.

[80] M. Pal, "Random forest classifier for remote sensing classification," *International Journal of Remote Sensing*, vol. 26, no. 1, pp. 217–222, 2005.