ConfSys 4: An Advanced Conference Management System with Automatic Semantic Header Generation

Yogesh O. Yadav

A Thesis

in

The Department

of

Computer Science and Software Engineering(CSSE)

Presented in Partial Fulfillment of the Requirements

for the Degree of

Master of Computer Science (Computer Science) at

Concordia University

Montréal, Québec, Canada

January 2024

CONCORDIA UNIVERSITY

School of Graduate Studies

This is to certify that the thesis prepared

By: **Yogesh O. Yadav**

Entitled: **ConfSys 4: An Advanced Conference Management System with Automatic Semantic Header Generation**

and submitted in partial fulfillment of the requirements for the degree of

Master of Computer Science (Computer Science)

complies with the regulations of this University and meets the accepted standards with respect to originality and quality.

Signed by the Final Examining Committee:

_____ Chair
Dr. Sabine Bergler

_____ Examiner
Dr. Nematollaah Shiri V.

_____ Supervisor
Dr. Bipin C. Desai

Approved by _____
Paquet, Joey, Chair
Department of Computer Science and Software Engineering(CSSE)

_____ 2024

_____
Mourad Debbabi, Dean
Faculty of Engineering and Computer Science

Abstract

ConfSys 4: An Advanced Conference Management System with Automatic Semantic Header Generation

Yogesh O. Yadav

ConfSys, a conference management system, has been used for over 15 years to manage a number of international academic conferences, such as C3S2E, IDEAS, and ICCM. It supports multi-event, multi-track conferences with a large number of user participation and submissions. It provides essential services such as setting up a conference, user sign-ups, call for papers, paper submissions, paper auction, paper allocation and review, blind debate, paper decision, author registration, final version submissions, automatic session management, presentation uploads, program generation, managing event sessions and electronic proceedings creation to efficiently manage and support the running of academic conferences and journals.

This thesis presents the fourth iteration in the ConfSys system to further accelerate enhancements and to incorporate new features keeping in sync with recent technological advancements. It presents new approach for Automatic Semantic Header Generation (ASHG 2) in Information Retrieval from academic documents such as research papers. ConfSys4 includes modules for document processing, information retrieval, and document classification. Document processing involves conversion of PDF documents to XML-formatted documents. Information retrieval involves extraction of paper-related details such as title, abstract, keywords, author names, emails, organizations, locations, affiliations, and author references present within the document. The extraction of author-related details ensures verification of the author metadata and references section for the submitted document. Document classification involves extraction of important subject headings (keywords) based on the contents of the submitted document. Thus, improving the paper submission, single/double/triple-blind review, and paper allocation process by reducing manual data entry

by users. Additionally, ConfSys4 includes improvements to existing features, such as adding automatic reminder emails to program committee members for updating their topics of interest for improved paper allocation and using entity matching technique for author pairs identification with conflicting interests. Furthermore, the PayPal payment interface is improved to include a standard checkout feature for payments and PDF document generation for the final program, invoice, and payment receipt for user registration to events. These improvements in ConfSys4 ensure consistent metadata generation for papers, improved transparency, ease of usability, and operability for organizing committee members, authors, and system users of the ConfSys system.

Acknowledgments

I would like to express my most sincere thanks to the following. First and foremost, to my thesis supervisor Dr. Bipin C. Desai for his constant support, guidance, mentorship, and patience throughout my journey.

Secondly, I would like to acknowledge the amazing work done by previous developers of ConfSys including Ming Lu, Min Huang, Kunsheng Zhao, and Yuwei Feng. Their work in building a strong foundation for ConfSys over the years really helped me in understanding the system's functionality and its internal workings.

To all the faculty, staff, and Database lab members at Concordia University, I would like to express my deepest appreciation for believing and trusting in my journey over the past two years. Special thanks to my Master's colleagues Akshay Dhabale, Pratik Bagora, Mrinal Rai, Siddhartha Jha and Kshitij Yerande for their constant support and motivation.

Finally, I would like to show my gratitude to my parents Omprakash and Asha Yadav, elder brother, and sister Ajit and Anita Yadav, extended family members, Divya Sharma, and close friends for their constant faith, support, and presence throughout this journey.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Overview

Academic research has played an important role in the advancement of technology for mankind. It has become a medium for exploring new ideas, problem-solving, and building solutions for knowledge expansion, peer validation, and innovation. In the last few decades, with the rise of the internet and the World Wide Web (www), systems for organizing academic conferences have emerged. These systems assist researchers in addressing industry needs, promoting information sharing, encouraging collaboration, and making information easily available. Prior to these systems, conventional tools and methods for academic research were physical libraries, printed journals, physical conferences with postal support systems, and traditional publishing. These methods had several limitations, such as low accessibility due to geographic constraints, limited global reach and peer reviews, and no centralized repository, making it challenging to keep up-to-date with the latest advancements, thus impacting the speed of technological growth.

Conference Management Systems (CMS) is a transformative and emerging solution to the limitations of traditional methods. These systems are digital platforms designed to streamline the organization and dissemination of research presented at conferences. They offer several benefits, including:

- Efficient Management - The organization of a conference requires assigning key roles of

general chair, program chair, track chair, and program committee members. Delegating responsibilities to manage and support conference-related tasks such as setting up an event, abstract and full paper submissions, paper auction and allocation, paper review, blind debate, paper decision, author registration, final version submissions, session management, and program generation. These roles and responsibilities work hand-in-hand and are automated for the smooth functioning of the event managed by a conference management system

- Higher Participation - CMS is a digital platform accessible to anyone over the internet. The reach of these systems encourages global participation and contribution to the conference and research community
- Central Repository - eproceedings are created and available over the internet is accessible to anyone at any time
- Discussion and Collaborative platform - These systems encourage interaction between authors and reviewers to exchange feedback to improve the quality of the submissions
- Networking - It encourages social interaction among its participants and enables collaboration beyond the conference

ConfSys [1], [2], [3] is one such system for managing academic conferences. It is a digital platform that has continuously improved to efficiently organize, manage, support, and deliver successful conferences for over 15 years. Conference events of size ranging from small group (up to 50-75) submissions, medium (up to 200-300) to large series (up to 1500-2000) with multiple tracks can be easily set up and managed completely on Confsys. Each such series can import the program committee and other details from a previous occurrence or add them directly using the custom-designed template. The users are able to use a single login credential for all events managed by ConfSys. The system manages all the essential functionalities, including those required for authors, program committee members, general chair, program chair, track chair, local chair, reviewers, and front desk.

The foundation of ConfSys [4] as a web application system for managing conferences was laid out by Dr Bipin C. Desai in the early 2000s. Over the past two decades, improvements have been incorporated into ConfSys with three stable system releases. With each release, the technology

stack was upgraded, existing functions were improved, and new features were added based on the past experiences of the managed events to keep the platform in sync with market demands.

The ConfSys4 system presented in this thesis is responsible for the implementation of modules for document processing, information retrieval, and document classification. These modules, in turn, are used for the implementation of the metadata extractor system known as Automatic Semantic Header Generation (ASHG 2). The document processing module is developed using third-party libraries such as PDFMiner and LXML for PDF document conversion to XML-formatted files. The information retrieval and document classification module is developed using language processing frameworks such as Natural Language Took Kit (NLTK), Spacy, BERT Transformers, classification algorithms such as Conditional Random Field (CRF) and Naive Bayes (NB) for metadata extraction, and subject headings (keyword) classification. These modules are developed using Python-supported libraries and integrated into a web development framework implemented in Java. It is hosted on an Apache-Tomcat server (version 9) with open-source MariaDB as the database engine. ConfSys4 transforms, modernizes, and advances the existing ConfSys3.5 system capability with the implementation and integration of these modules to enable metadata extraction and classification using ASHG 2 from PDF documents.

1.2 Areas of Improvement

1.2.1 User Dependency in Paper Submission

Authors submit their research papers during the Call For Paper (CFP) period. During the submission, author is required to manually enter paper-related details such as title, abstract, keyword, co-author details, subject hierarchy selection, and upload a PDF file of the paper. The author won't be able to submit a paper if any of the above details are missing. Paper submission by author has human dependency and the possibility of errors such as entering incorrect information for title, author/co-author details, abstract, keywords, and subject hierarchy selection. In addition, subject hierarchy selection is subjective, may vary from person to person, and could impact the paper allocation process. Furthermore, author can upload the same PDF file multiple times or an invalid PDF as a submission entry. In ConfSys3.5 system [3], the uploaded PDF file is not processed to identify

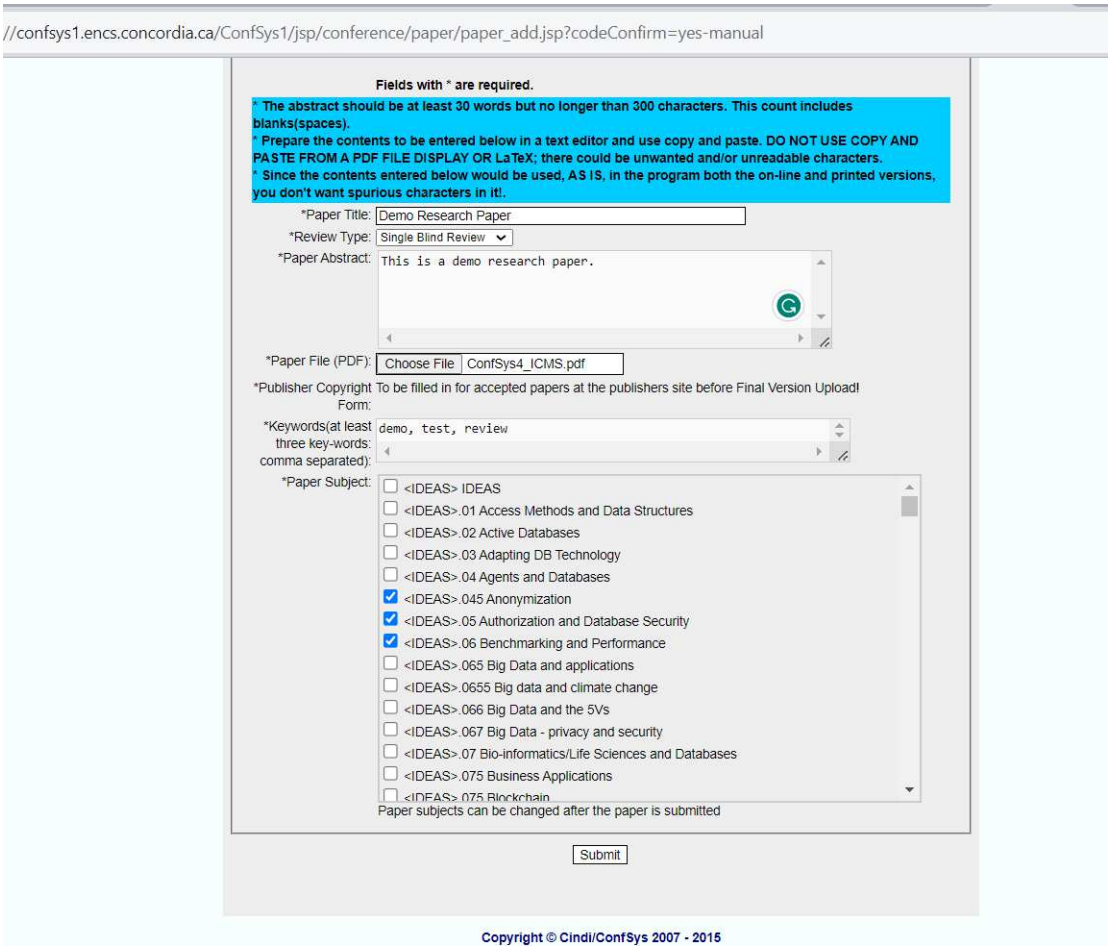

Figure 1.1: Paper Submission Event - Manual Data Entry

duplicate paper or invalid PDF submissions. The manual data entry by the author and document processing to eliminate errors and avoid duplicate paper or invalid PDF submissions is identified as an area of improvement for the system.

Fig 1.1 shows the data entry screen provided to the author for adding paper details and uploading PDF documents. Fig 1.2 shows the alert notification raised for incorrect data entry by the author in the ConfSys3.5 system [3].

1.2.2 Reminder to Program Committee Members for Topic of Interest

Another essential feature of ConfSys3.5 [3] is to automatically allocate submitted papers to program committee members for initial bidding of papers. To make an initial allocation, the system

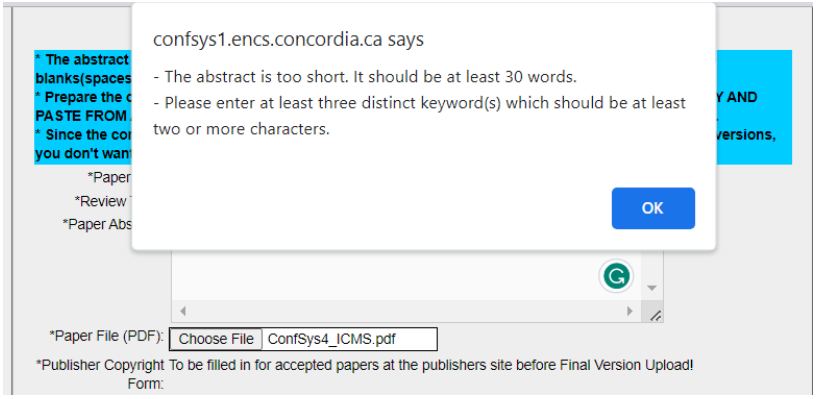

Figure 1.2: Paper Submission Event - Alert Raised to Submitting Author

tries to match the subject headings (keywords) associated with the papers to the program committee member's topics (keywords) of expertise. The chairs could modify this allocation if required.

We have found that this allocation function works well if:

- The members of the program committees have completed their topics of interest choices, and
- The authors have selected appropriate subject headings (keywords) for their paper

However, as illustrated in Fig. 1.3, the system faces two problems:

Most of the program committee members do not complete their topics of interest. Also, the subject headings (keywords) of the papers may not be correctly selected by the authors. Without this information, the system, at best, assigns papers to these reviewers at random. However, since ConfSys tries to find matches before resorting to this random process, some reviewers with completed topics of interest are overloaded, while others without any are not assigned any papers to bid for and, hence, to review. In order to avoid overloading, program committee members need to be regularly reminded to update their topic of interest. The automatic reminder emails to program committee members for updating topics of interest is considered as an area of improvement for the system.

1.2.3 User Dependency for Single/Double/Triple Blind Review in Paper Submission

In ConfSys3.5 [3], the author manually selects the review type as single/double-blind during the paper submission process, as shown in Fig 1.1. In the case of double-blind selection, the system

Paper Allocation Match Rate

Reviewer Interests Match Rate

Total number of reviewers : 105

Reviewer ID	Total Number of Reviewer's Interests	Total Number of Allocated Papers	Number of Allocated Paper of which the topics match reviewer's interests	Match rate(%)
6	0	1	0	No Interests
10	0	5	0	No Interests
18	0	1	0	No Interests
20	0	1	0	No Interests
24	0	1	0	No Interests
25	16	2	2	100
30	0	1	0	No Interests
33	0	1	0	No Interests
44	11	1	1	100
51	0	1	0	No Interests
52	0	0	0	No Interests
59	0	1	0	No Interests
67	8	5	2	40
77	0	3	0	No Interests
82	0	2	0	No Interests
208	0	1	0	No Interests
232	6	0	0	No Match
295	10	2	2	100
311	0	1	0	No Interests
313	0	1	0	No Interests
322	0	1	0	No Interests
324	14	2	1	50
402	33	2	2	100
416	16	2	2	100

Figure 1.3: Reviewers Interests and Assigned Papers Match Rate

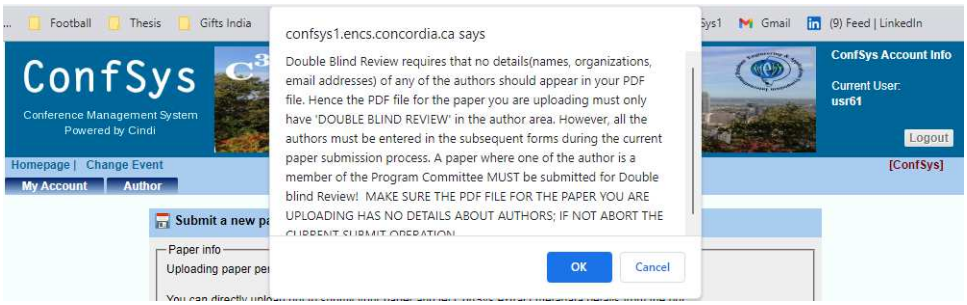

Figure 1.4: Double-blind Alert Message in ConfSys

raises an alert message to the author to exclude author details in the metadata section and references section of the uploaded PDF file if the paper authors belong to the organizing committee to maintain anonymity in the review process as shown in Fig 1.4. If one of the paper authors is a chair (Track Chair/Program Chair/General Chair), the paper must be submitted as double-blind, and the system will convert it to triple-blind. The author can select double-blind and upload a PDF file with author details in the metadata section or references section either by mistake or by ignoring the alert message. The ConfSys3.5 system [3] does not verify the PDF contents to prevent authors from uploading PDF files containing author-related details. It is identified as an area of improvement for the system.

1.2.4 Author Pair Identification for Conflict-of-Interest Resolution

In ConfSys3.5 [3], author pairs identification with conflicting interests is carried out by exact matching of author names using the publicly available DBLP dataset [5]. The DBLP dataset contains bibliographic information about computer science publications and includes metadata details such as titles, authors, publications, abstracts, and keywords associated with the academic research paper. The author names are extracted and consolidated for each paper in the DBLP data set, author pairs are generated and matched with system user names. If matched for both authors, the pair is linked to any past papers and considered to have conflicting interests. The exact author name matching can misidentify potential author pairs with minor differences in their names. Improvement in the author name match process to avoid minor author name differences is considered as an area of improvement for the system.

1.2.5 PDF Document for Invoice Receipt of User Registration and Generated Program

In ConfSys3.5 [3], users are allowed to register for conferences by paying the registration fees. The registered users do not receive any invoice and payment receipt as a confirmation. In addition, the General Chair/Admin creates the preliminary or final program, but there is no option available to generate a PDF document of the created program schedule and download it for sharing with registered users. The PDF document generation for invoice, payment receipt and preliminary or final program is considered as an area of improvement for the system.

1.2.6 Update Payment interface for PayPal

The ConfSys3.5 system supports payment integration with widely used third-party application software such as Paypal and Moneris. PayPal has released new features such as standard checkout to ensure easy, safe, secure, and reliable payments since the release of ConfSys3.5. Improving the existing payment interface for PayPal mode is considered as an area of improvement for the system.

1.3 ConfSys4

Over the past decades, computer's capability of computing, processing, and classifying text documents, multimedia objects such as images and videos, and speech across a large corpus of data has seen considerable advances. The computers are capable of processing large volumes of information, segregating words, sentences, and paragraphs, summarizing the text, drawing inferences, extracting semantic relationships, and language translations. This is possible with recent advances in natural language processing of unstructured human language data, document processing, and information extraction. Thus, building a module to reduce user data entry during the paper submission process is now possible by extracting the underlying metadata from the submitted PDFs. Furthermore, it is also possible to analyze the file's contents and classify the paper to appropriate subject headings (keywords).

The ConfSys4 system extracts metadata details such as title, keywords, abstract, author details, paper body contents, and author references using the metadata extractor system (ASHG 2). It

also associates appropriate subject headings (keywords) based on the contents of the papers. The submitting author is required to verify and update the extracted metadata and subject heading classifications before submitting the paper. In addition, the system verifies the uploaded file for author metadata and references involving the paper authors. Furthermore, the system periodically executes reminder routines to send reminders to the program committee members to update their topics of interest until the auction period is reached. The goal of the ConfSys4 system is to improve the paper submission, single/double-blind review, paper allocation process, and, in turn, the quality of the publication.

Extracting relevant semantics from an unstructured text document is a known research topic [6–16] in the scientific community. The methodologies and techniques for extracting semantics can be designed as a simple rule-based system [10], [11], [9] where heuristics are defined for extracting metadata based on pattern matching and logical structure of the document. These systems perform well for documents with fixed layouts for metadata extraction. Another approach to leverage supervised classification machine learning techniques such as Naive Bayes, Logistic Regression, Conditional Random Fields (CRF), Support Vector Machines (SVM), and Hidden Markov Model (HMM) to train and test the model on document contents and labeled metadata. The model learns features such as patterns, relationships, and dependencies during the training phase and is able to predict metadata labels for the new submitted document as input. Extraction of metadata information for the submitted document is possible by employing these classification techniques. These techniques produce consistent results and generalize well to a variety of research document layouts. The extractor systems such as CERMINE [12], [13], and GROBID [14] are examples of these machine learning-based techniques.

The ConfSys4 implemented extractor system ASHG 2 uses a hybrid combination of rule-based heuristics, supervised classification machine learning techniques such as Conditional Random Field (CRF) and Naive Bayes (NB), natural language processing techniques such as text normalization, and named entity recognition for extraction of metadata elements and classification to subject heading (keywords) from the input PDF documents.

1.4 Organization of Thesis

The thesis is structured as follows.

- Chapter 2 describes existing work in Automatic Semantic Header Generator (ASHG 1), Conference Management systems such as ConfSys3.5, and other competitors such as EasyChair, ConfTool, OpenConf, and Oxford Abstract.
- Chapter 3 describes the technology stack added to the ConfSys4 system.
- Chapter 4 describes the new features and improvements to existing functionalities in the ConfSys4 system.
- Chapter 5 describes the implementation of the metadata extraction module and its integration with the Confsys system.
- Chapter 6 describes the experimentation and evaluation results for the ConfSys4 system.
- Chapter 7 describes thesis contributions and discusses future work for the ConfSys system.

Chapter 2

Related Work and Existing CMS Systems

2.1 Automatic Semantic Header Generator (ASHG 1)

The Concordia INdexing Discovering System, CINDI in short [17], a digital library system built to provide a unified platform for academic communities to browse, explore, upload, and annotate academic and scientific documents. It introduced the notion of having a metadata descriptor known as Semantic Header [18] for an information resource. The Semantic Header and techniques for the automatic and reliable generation of Semantic header, proposed in [19], [20], [21], [22] attempt to capture the semantics of information resources and relationships among different sections in a given information item.

Automatic Semantic Header Generator (ASHG 1) is a metadata extractor tool developed with the goal of automatically building reliable semantic header information that provides the capability of easy search and access to an information resource. The semantic header includes important metadata details such as title, subject, abstract, keywords, author details, and subject classifications about an information resource. These metadata details are most often used to search an information document over a large corpus of available resources. It was developed to replace traditional systems extracting semantic information using keyword-based methods. These keyword-based systems attempt to search specified keyword terms inside the information resource without the context of the

target information resource, thus providing a major obstacle to the accurate retrieval of metadata details for an information resource. As described in the ASHG paper [9], semantic header information has two main advantages over the traditional keyword-based indexing methods :

- Represent information resources more completely by including items such as title, author, keywords, subject, date, genre, subject classification, coverage, etc.
- Support to data extraction methods used by search engines.

Steps Involved in ASHG 1

The major steps involved in building Semantic Header information as described in the paper [9] are as follows :

(1) Document Type Recognition - recognize the document type submitted based on the file naming conventions (PDF, HTML, LATEX).

(2) Extractor Application - extract metadata elements such as title, author, author details, keywords, abstract, creation date, and subject using the rule-based heuristic function.

(3) Document Classification - assigning a list of subject headings to the document. It involves steps such as removing stopwords (common English words) occurring frequently in a document, performing a stemming process to map these extracted words to a base root word, and finally looking to the controlled term subject dictionary for generating the list of implicit keywords for the document.

(4) Header Validation - extracted metadata details and a list of subject headings are shared with the user for validation and review.

2.2 Existing Systems

This section describes existing conference management systems (CMS), their support to baseline services, and roles for efficient management of conference events. A comparison of features

supported by the ConfSys4 vs. ConfSys3.5 and other peer CMS systems is described at the end of the section.

2.2.1 ConfSys 3.5

The ConfSys 3.5 is the existing stable version of the ConfSys platform, managing and hosting conferences for the past decade. The system was developed by Ming Lu under the supervision of Dr. Bipin C Desai. ConfSys 3.5 described in the paper [3] added several essential features such as single/double/triple-blind review, auto session management, front desk role, eproceedings generation, and program generation, improved existing features such as new user signup, user email, registration integration, key usability dimensions such as menu bar and search function in paper management, paper submission, and editing significantly improved the performance, usability, and user experience of the ConfSys system. The platform was upgraded from Java Servlet Programming to Servlet/Struts 2 hybrid to handle peak traffic load on the system. ConfSys3.5 is hosted on Apache Tomcat and uses an open-source MariaDB database engine similar to previous versions. It provides all essential solutions and services to support various roles in managing academic conferences. It supports roles such as Administrator, General Chair, Program Chair, Program Committee, Track Chair, Local Chair, Reviewer, Front Desk, and Author to perform operations such as flexible event configurations, paper submission, paper auction, paper auto-allocation, paper review, single/double/triple-blind review, blind debate, paper decision, auto session arrangement, and session support, program generation, registration integration, and eproceeding creations.

Organizing and managing a conference event on ConfSys3.5 typically involves

(1) Event Setup - Set up milestones for the event, set up a committee for the event such as General Chair, Program Chair, and Program Committee, and set up of subject hierarchy for the event.

(2) User Signup - In order to access ConfSys, a user needs to sign up for the first time using the email address. The user could then login into the system using the shared credential, update the basic profile information, and select areas of interest from the list of topics presented by the system in order to have access to the ”Author” role to submit a paper.

(3) Author Submission - The author manually enters metadata details related to the paper such

as title, abstract, keyword, review type, and subject hierarchy selection, and uploads a PDF document before submitting the paper for an event during the Call for Paper (CFP) period.

(4) Automatic Paper Allocation for Review - The system automatically allocates paper to program committee members based on matching the paper's topic of interest with the member's topic of expertise for initial bidding during auction and for paper review.

(5) Double Blind Debate for Controversial Paper - System organizes debate among the reviewers for controversial papers. In double-blind, reviewers do not know each other's identity.

(6) Paper Decision, Registration, and Final Version Management - The program/track chair makes the final decision for accepting/rejecting the paper based on the acceptance criteria and scores assigned to the paper. Accepted Authors need to register for the event and submit the final version of the paper.

(7) Slide Uploads - Authors need to upload presentations for the event program into the system.

(8) Auto Session Management, Preliminary and Final Program - the system offers auto session management for generating the preliminary or final program with little manual involvement.

(9) eProcceeding Creation - eproceedings, a digital document, the preferred option for both organizers and readers, is created for the event.

2.2.2 Other CMS Systems

EasyChair

EasyChair (EC) [23] is also a web-based conference management and peer-review software. It has been used since 2002 in the scientific community for tasks such as organizing research paper submissions and reviews. It was designed to help conference organizers cope with the complexity of the refereeing process. The latest version of EasyChair was released on May 24, 2018. It offers free, professional, and executive services for organizing small, medium, and multi-event multi-track conferences. EasyChair supports calls for submissions, abstract/paper submissions, reviewer management, paper review and decision, program editing and publishing, conference proceedings

generation, and attendee registration for management of the event. Additional Services supported by EasyChair :

- Smart CFP for publishing calls for papers for the new events on their webpage.
- Smart Slide for publishing conference presentations.
- Publishing conference proceedings.
- Conference registration, with or without online payment.

ConfTool

ConfTool [24] is another Web-based event management software developed to support the organization of academic conferences, workshops, congresses, and seminars. ConfTool features include the paper submission and review process, scheduling of the conference program, registration, administration, invoicing of participants, facilitation of communication between authors and participants, and availability in over 15 languages. ConfTool provides two versions: standard and professional, depending on the event size, services included, and the cost of organizing an event. Benefits of using ConfTool :

- Customizable submission and review forms.
- Scheduling of the conference program with access to sessions, presentations, and accepted contributions.
- Flexible forms for participation registration with payment options, including PayPal.
- General Data Protection Regulation (GDPR) Compliant.

OpenConf

OpenConf [25] is a Peer-Review, Abstract, and Conference Management software application originally used for conferences only, but it has been adapted for use by events such as journals, workshops, books, symposia, grants, and competitions. OpenConf is available in multiple editions, including Community (Free - small conferences), Plus (medium conferences), and Professional

(conferences of all sizes). The software is available for download and cloud-based, with translations in over a dozen languages for author and reviewer interfaces. Developed using PHP and a MySQL database backend, the downloaded version of OpenConf runs on various platforms, including Linux, Mac OS X, and Windows. An HTML5-compliant mobile app is also available for use by event participants accessing the program schedule. Services included by OpenConf :

- Custom Forms for submission, review, and committee profile.
- Automatic Paper Allocation to reviewers for submitted papers.
- Program Generation to facilitate building and publishing of online program.
- Communication between committee members and authors through online discussion is provided to help improve the final paper or presentation.
- User registration payment support for events

Oxford Abstracts - Conference Management Software

Oxford Abstracts [26] is a conference management software tailored to academic events. It was founded in 2001 and is currently headquartered in Scotland. Several products, such as abstract management, conference management, award management, and registration delegation, are provided by Oxford Abstract. It provides services such as

- submission for abstract, complete paper, applications, or proposals.
- easy review process for the reviewer, paper decision with flexible acceptance types for submitted papers.
- multistage events support for large events.
- eproceedings generation for the event.

2.2.3 ConfSys4 vs Other CMS systems

Over the past two decades, several conference management systems have been developed to adapt to the current demands in the market. ConfSys3.5, EasyChair, ConfTool, OpenConf, and

Oxford Abstracts described in the previous section are CMS platforms that have organized and managed a number of successful academic events for several years. These systems have gradually transformed their processes to manage conferences and cater to the world's needs. ConfSys3.5, described earlier, is a platform with most of the features compared to its peers for managing small or large conferences with functionality and roles defined for most of the roles involved.

ConfSys4 incorporates document processing, information retrieval, and document classification modules for processing the document, extracting salient metadata details, and classifying to appropriate subject headings (keywords) from the submitted document, in addition to supporting all services and roles provided by ConfSys3.5. Table 2.1 shows features support comparison across ConfSys4, ConfSys3.5, and other CMS systems. No other CMS system has yet incorporated the extraction of metadata details and subject headings classification from PDF documents. Thus, metadata extraction, author details identification and appropriate selection of subject heading (keyword) are features supported by ConfSys4. Other systems have focused on making their system adapt to more languages, making it accessible worldwide and encouraging academic work not to have the constraint of language. This is currently a drawback of ConfSys platforms 3.5 and 4, which could be considered in future work to encourage conference organizers to consider ConfSys for multilingual conferences.

Features/Functions	**EasyChair**	**ConfTool**	**OpenConf**	**Ox. Abstracts**	**ConfSys3.5**	**ConfSys4**
Conference Creation and Multi Track Support	Y	Y	Y	Y	Y	Y
Hosting and Tech Support	Y	Y	Y	Y	Y	Y
Paper/Abstract Submission	Y	Y	Y	Y	Y	Y
Metadata Extraction in Paper Submission	N	N	N	N	N	**Y**
Author Detail Extraction from Metadata and References Section for Single, Double, Triple Blind Review	N	N	N	N	N	**Y**
Custom Topic Selection and Auto Tagging in Paper during submission	N	N	N	N	N	**Y**
Multilingual	N	Y	Y	N	N	N
Multiple Payment options	Y	Y	N	Y	Y	Y
Automatic/Manual Paper Allocation	Y	Y	Y	Y	Y	Y
Paper Auction and Review	Y	Y	Y	Y	Y	Y
Single and Double Blind Review	Y	Y	Y	Y	Y	Y
Triple Blind Review	N	N	N	?	Y	Y
Program Creation	Y	Y	Y	Y	Y	Y
Automatic Session Management	Y	Y	N	?	Y	Y
eProceeding Creation	Y	N	N	Y	Y	Y
Conflict of Interest Checking	Y	Y	?	?	Y	Y
Participant Registration	Y	Y	N	Y	Y	Y
Invoice Generation	Y	Y	N	Y	N	Y
Double Blind Debate for Controversial Paper	Y	?	?	?	Y	Y
Communication and Messaging System	Y	Y	Y	?	Y	Y

Table 2.1: Comparison of ConfSys4 with Other CMS Systems Easy Chair, ConfTool, OpenConf, Ox. Abstracts and ConfSys3.5 - Y, N, and ? represent feature support, no feature support, and no clear information available about feature support

Chapter 3

Technology Stack

3.1 System Overview

ConfSys4 is a web-based platform accessible over the internet. All system modules are integrated and function as one system, thus hiding internal complexity, communication, and functions from the external user. It is hosted on the open-source Apache-Tomcat server (version 9), developed using java servlets/struts hybrid web framework, dynamic jsps, jquery, asynchronous javascript and xml (ajax), html/css, and uses open-source MariaDB as the database engine similar to previous versions of Confsys [1–3]. This thesis implements modules for document processing, information retrieval, and document classification. The libraries such as scikit-learn [27], Natural Language Tool Kit [28], Spacy [29], Hugging Face [30], BERT Transformer Models [31, 32], Regular Expressions [33], LXML [34] and PDFMiner [35] are used for the implementation of these modules. These modules are developed in python language and integrated into the java web framework to incorporate metadata extraction and classification from PDF documents.

3.2 Document Processing Module

Document Processing Module involves 1) the conversion of PDF document to raw XML format document using the library PDFMiner [35], and 2) the conversion of raw XML document to pre-processed XML document using the library LXML [34]. The pre-processed XML document is then

C: > Users > yoges > PycharmProjects > WebScrapped_data > xml > ConfSys_Intelligent_CMS_ACM.xml

```
<?xml version="1.0" encoding="utf-8" ?>
<pages>
<page id="1" bbox="0.000,0.000,612.000,792.000" rotate="0">
<textbox id="0" bbox="81.382,692.329,530.618,714.537">
<textline bbox="81.382,692.329,530.618,714.537">
<text font="ECCUWF+LinBiolinumTB" bbox="81.382,692.329,93.536,714.537" size="22.208">C</text>
<text font="ECCUWF+LinBiolinumTB" bbox="93.536,692.329,103.280,714.537" size="22.208">o</text>
<text font="ECCUWF+LinBiolinumTB" bbox="103.280,692.329,113.609,714.537" size="22.208">n</text>
<text font="ECCUWF+LinBiolinumTB" bbox="113.609,692.329,119.910,714.537" size="22.208">f</text>
<text font="ECCUWF+LinBiolinumTB" bbox="119.910,692.329,129.241,714.537" size="22.208">S</text>
<text font="ECCUWF+LinBiolinumTB" bbox="129.241,692.329,138.520,714.537" size="22.208">y</text>
<text font="ECCUWF+LinBiolinumTB" bbox="138.520,692.329,145.733,714.537" size="22.208">s</text>
<text> </text>
<text font="ECCUWF+LinBiolinumTB" bbox="150.037,692.329,155.959,714.537" size="22.208">-</text>
<text> </text>
<text font="ECCUWF+LinBiolinumTB" bbox="160.263,692.329,172.331,714.537" size="22.208">A</text>
<text font="ECCUWF+LinBiolinumTB" bbox="172.331,692.329,182.660,714.537" size="22.208">n</text>
<text> </text>
<text font="ECCUWF+LinBiolinumTB" bbox="186.964,692.329,192.645,714.537" size="22.208">I</text>
<text font="ECCUWF+LinBiolinumTB" bbox="192.645,692.329,202.974,714.537" size="22.208">n</text>
<text font="ECCUWF+LinBiolinumTB" bbox="202.974,692.329,209.069,714.537" size="22.208">t</text>
<text font="ECCUWF+LinBiolinumTB" bbox="209.069,692.329,217.814,714.537" size="22.208">e</text>
<text font="ECCUWF+LinBiolinumTB" bbox="217.814,692.329,223.048,714.537" size="22.208">l</text>
<text font="ECCUWF+LinBiolinumTB" bbox="223.048,692.329,228.281,714.537" size="22.208">l</text>
<text font="ECCUWF+LinBiolinumTB" bbox="228.281,692.329,233.652,714.537" size="22.208">i</text>
<text font="ECCUWF+LinBiolinumTB" bbox="233.652,692.329,243.310,714.537" size="22.208">g</text>
<text font="ECCUWF+LinBiolinumTB" bbox="243.310,692.329,252.055,714.537" size="22.208">e</text>
<text font="ECCUWF+LinBiolinumTB" bbox="252.055,692.329,262.385,714.537" size="22.208">n</text>
<text font="ECCUWF+LinBiolinumTB" bbox="262.385,692.329,268.479,714.537" size="22.208">t</text>
<text> </text>
<text font="ECCUWF+LinBiolinumTB" bbox="272.783,692.329,284.937,714.537" size="22.208">C</text>
<text font="ECCUWF+LinBiolinumTB" bbox="284.937,692.329,294.681,714.537" size="22.208">o</text>
<text font="ECCUWF+LinBiolinumTB" bbox="294.681,692.329,305.010,714.537" size="22.208">n</text>
<text font="ECCUWF+LinBiolinumTB" bbox="305.010,692.329,311.311,714.537" size="22.208">f</text>
<text font="ECCUWF+LinBiolinumTB" bbox="311.311,692.329,320.056,714.537" size="22.208">e</text>
<text font="ECCUWF+LinBiolinumTB" bbox="320.056,692.329,327.304,714.537" size="22.208">r</text>
<text font="ECCUWF+LinBiolinumTB" bbox="327.166,692.329,335.912,714.537" size="22.208">e</text>
```

Figure 3.1: Sample PDF to XML Conversion

passed as an input to the information retrieval module.

3.2.1 PDF-To-XML Conversion

The ASHG 1 [9] used third-party converter Xpdf-2.0.1, an open-source software package to convert PDF format documents to text-formatted documents. The extractor ASHG 2 implemented in ConfSys4 required additional features such as layout and position-based details in addition to textual information about the information resource. This led to using the PDFMiner [35] library for converting PDF to XML-formatted documents. PDFMiner is a third-party python library offering a command-line utility tool to directly convert PDF documents to XML or Text-based formats.

As illustrated in Fig 3.1, the PDFMiner library converts the PDF document into an XML-formatted document. Additional details such as the geometric location (bounding box), relative positive (X, Y), and layout features (font size and font style) associated with XML node elements are readily available and can be leveraged further for information extraction.

C: > Users > yoges > PycharmProjects > WebScrapped_data > output > modified_modified_ConfSys_Intelligent_CMS_ACM.xml

```xml
<?xml version="1.0" encoding="UTF-8"?>
<pages>
    <page id="1" bbox="0.000,0.000,612.000,792.000" rotate="0">
        <textbox id="0" bbox="81.382,692.329,530.618,714.537" font="ECCUWF+LinBiolinumTB" size="22.208">
            ConfSys - An Intelligent Conference Management System
        </textbox>
        <textbox id="1" bbox="143.265,672.558,223.460,689.080" font="JZEECJ+LinLibertineT" size="16.522">
            Yogesh O. Yadav
        </textbox>
        <textbox id="2" bbox="124.288,660.603,242.484,674.371" font="JZEECJ+LinLibertineT" size="13.768">
            yogeshoyadav08@gmail.com
        </textbox>
        <textbox id="3" bbox="139.949,648.647,227.082,662.415" font="JZEECJ+LinLibertineT" size="13.768">
            Concordia University
        </textbox>
        <textbox id="4" bbox="129.573,636.692,237.199,650.460" font="JZEECJ+LinLibertineT" size="13.768">
            Montreal, Quebec, Canada
        </textbox>
        <textbox id="5" bbox="394.803,672.558,464.418,689.080" font="JZEECJ+LinLibertineT" size="16.522">
            Bipin C. Desai
        </textbox>
        <textbox id="6" bbox="375.717,660.602,483.502,674.370" font="JZEECJ+LinLibertineT" size="13.768">
            bipinc.desai@concordia.ca
        </textbox>
        <textbox id="7" bbox="386.173,648.647,473.306,662.415" font="JZEECJ+LinLibertineT" size="13.768">
            Concordia University
        </textbox>
        <textbox id="8" bbox="375.797,636.692,483.423,650.460" font="JZEECJ+LinLibertineT" size="13.768">
            Montreal, Quebec, Canada
        </textbox>
        <textbox id="9" bbox="53.529,526.545,295.562,629.076" font="JZEECJ+LinLibertineT" size="12.392">
            ABSTRACT
            This paper offers a brief history of, ConfSys, a conference man-
            agement system, that has been used for over 15 years to support a
            number of international academic conferences. The meeting span
            in size is measured by the number of submissions from small(from
            about 100, to 2000 - 3000). It is a complete system that has all
```

Figure 3.2: Preprocessed XML - Input to the model

3.2.2 XML Preprocessing

LXML [34] is a Python library for processing XML documents. It provides a simple and efficient way to parse, manipulate, and generate XML and HTML documents. LXML is built on top of the C libraries libxml2 and libxslt, which makes it fast and reliable for working with XML and HTML data.

In ConfSys4, the LXML library is used to convert raw XML to pre-processed XML. The pre-processing step consists of traversing XML node elements iteratively till the leaf node is reached, consolidating and transferring leaf node contents and features to its immediate parent node, and then eliminating the leaf nodes. Fig 3.1 shows the raw XML output of the PDF document, and Fig 3.2 shows the pre-processed XML output with leaf node ”text” contents consolidated to its immediate parent node, ”textbox”. This intermediate XML is passed as an input to the extractor module for information retrieval. Raw XML preprocessing reduces the size of the XML and traversal path for metadata extraction during the information retrieval step.

3.3 Information Retrieval and Document Classification Module

Information Retrieval and Document Classification Module involves 1) extraction of metadata objects such as title, abstract, keyword, author details, body contents, and author references, and 2) classification to subject headings (keywords) based on the contents of the information resource.

3.3.1 Classification with Machine Learning Algorithms

Scikit-learn [27] is an open-source machine-learning library for the Python programming language. It provides simple and efficient data analysis and modeling tools, including various machine-learning algorithms for classification, regression, clustering, and dimensionality reduction. It offers functionalities for data preprocessing, model evaluation, and model selection, making it a comprehensive package for building machine learning pipelines for prediction and classification tasks. In ConfSys4, the metadata extraction task is implemented using a Conditional Random Fields (CRF) sequence classifier model, and the classification to subject heading task is implemented using the Naive Bayes (NB) probabilistic classifier model.

3.3.2 Named Entity Recognition (NER)

Named Entity Recognition (NER) is a natural language processing (NLP) task to scan the unstructured text corpora to identify spans of text that constitute proper names and associate the type of proper name to the span of text. These identified spans of text are known as named entities, and the most common tag types used are PER(person), LOC(location), ORG(organization), or GPE(geo-political entity). NER is essential to many NLP applications, such as sentiment analysis, information retrieval, question answering, and machine translations. In academic articles, author details such as name, email, organization, location, and affiliation are present in the metadata section, and author citations are present in the references section. These author details and citations constitute the primary and secondary sources of author information in an information resource. Extraction and tagging of primary and secondary author details from an academic article helps in building a reliable author profile and, in turn, improves information retrieval for the information resource or the author-related data.

Natural Language Processing Toolkit [28] and Spacy [29], both libraries are widely used to work with human language unstructured data. Both libraries provide natural language processing functionalities like tokenization, stemming, lemmatization, normalization, part-of-speech tagging, named entity recognition, and text classification for various applications and tasks such as sentiment analysis, email spam detection, document classification, etc.

BERT(Bidirectional Encoder Representations from Transformers) is a transformer-based model [31] designed to learn features/relationships or dependencies from unlabeled text across both directions left and right given a context word for an input sequence. The pre-trained BERT model, with little fine-tuning, can achieve state-of-the-art results across a wide variety of natural language processing tasks such as named entity recognition, language inference, and translation. Hugging Face [30] provides an open-source platform for natural language processing (NLP) and machine learning tasks and provides support for BERT transformer models. Transformers [36], a state-of-the-art machine learning framework developed by hugging face, includes an implementation for PyTorch [37] and TensorFlow [38]. The framework provides APIs and tools to download and use state-of-the-art pre-trained models. These pre-trained models include BERT models(bert-base-ner, bert-large-ner) [31, 32], GPT models(GPT 2,3), RoBERTa, DistilBERT, etc. The use of pre-trained models helps to reduce the computing cost and save time and resources required to train a model from scratch.

In ConfSys4, the NER task for author identification and author details extraction for generating author profiles is implemented using libraries such as Natural Language Toolkit [28], Spacy [29], and Transformer BERT models(bert-base-ner and bert-large-ner) [31] from hugging face [30].

As stated in the Speech and Language book [39], ”One of the unsung successes in standardization in computer science has been the regular expression (RE), a language for specifying text search strings”. A regular expression search scans the entire corpus, returning all texts matching the search pattern. In ConfSys4, the regex Python library is used for pattern matching to identify various email address formats related to author metadata from the underlying PDF document.

```python

import Levenshtein

if __name__ == '__main__':
    string_a="John Doe"
    String_b="John D"
    print(f"String 1: {string_a}")
    print(f"String 2: {String_b}")
    print(f"Length of String 1: {len(string_a)}")
    print(f"Length of String 2: {len(String_b)}")
    print(f"Edit Distance Between two String: {Levenshtein.distance(string_a, String_b)}")
    print(f"Edit Distance Ratio Between two String: {Levenshtein.ratio(string_a, String_b)}")
```

```
Run  Edit-Distance-Ratio
C:\Users\yoges\AppData\Local\Programs\Python\Python311\python.exe C:\Users\yoges\Desktop\thesis\ConfSys4
String 1: John Doe
String 2: John D
Length of String 1: 8
Length of String 2: 6
Edit Distance Between two String: 2
Edit Distance Ratio Between two String: 0.8571428571428572

Process finished with exit code 0
```

Figure 3.3: Edit Distance Ratio Example

3.3.3 Similarity Ratio Score

The Levenshtein distance, also known as Edit Distance [40], is a metric to measure the similarity between two strings. It is defined as the minimum number of operations required to transform one string into another. These operations can be insertions, deletions, or substitutions of a single character. ConfSys4 uses the Levenshtein library available in Python to calculate the similarity ratio score for two input strings. The Levenshtein similarity ratio is computed as follows:

$$\text{Similarity Ratio} = \frac{(\text{Length of string1 + Length of string2}) - \text{Edit Distance(string1, string2)}}{\text{Length of string1+ Length of string2}} \quad (1)$$

It outputs similarity in the range of [0,1], 1 indicating identical or nearly identical strings, and 0 indicating dissimilarity between them. As seen in Figure 3.3, the Levenshtein distance between two strings is 2, and the sum of the length of both strings is (8+6=14). The similarity ratio calculated is (14-2/14) = 0.86, indicating a high similarity between the two input strings. The Levenshtein similarity ratio is used for matching author names and comparing metadata object strings for title,

abstract, and keyword during the metadata extraction phase. Furthermore, similarity ratio is used during the evaluation phase for comparing ConfSys4 system extractor results with ground-truth labeled metadata for metadata objects.

3.4 MVC Architecture

ConfSys4 is based on a Model-View-Controller(MVC) architecture. The modules highlighted in Fig 3.4 are incorporated with respect to the ConfSys3.5 version [3]. The addition of these modules in the Controller Layer facilitates metadata extraction from PDF documents and the classification to subject headings (keywords) based on the contents of the paper.

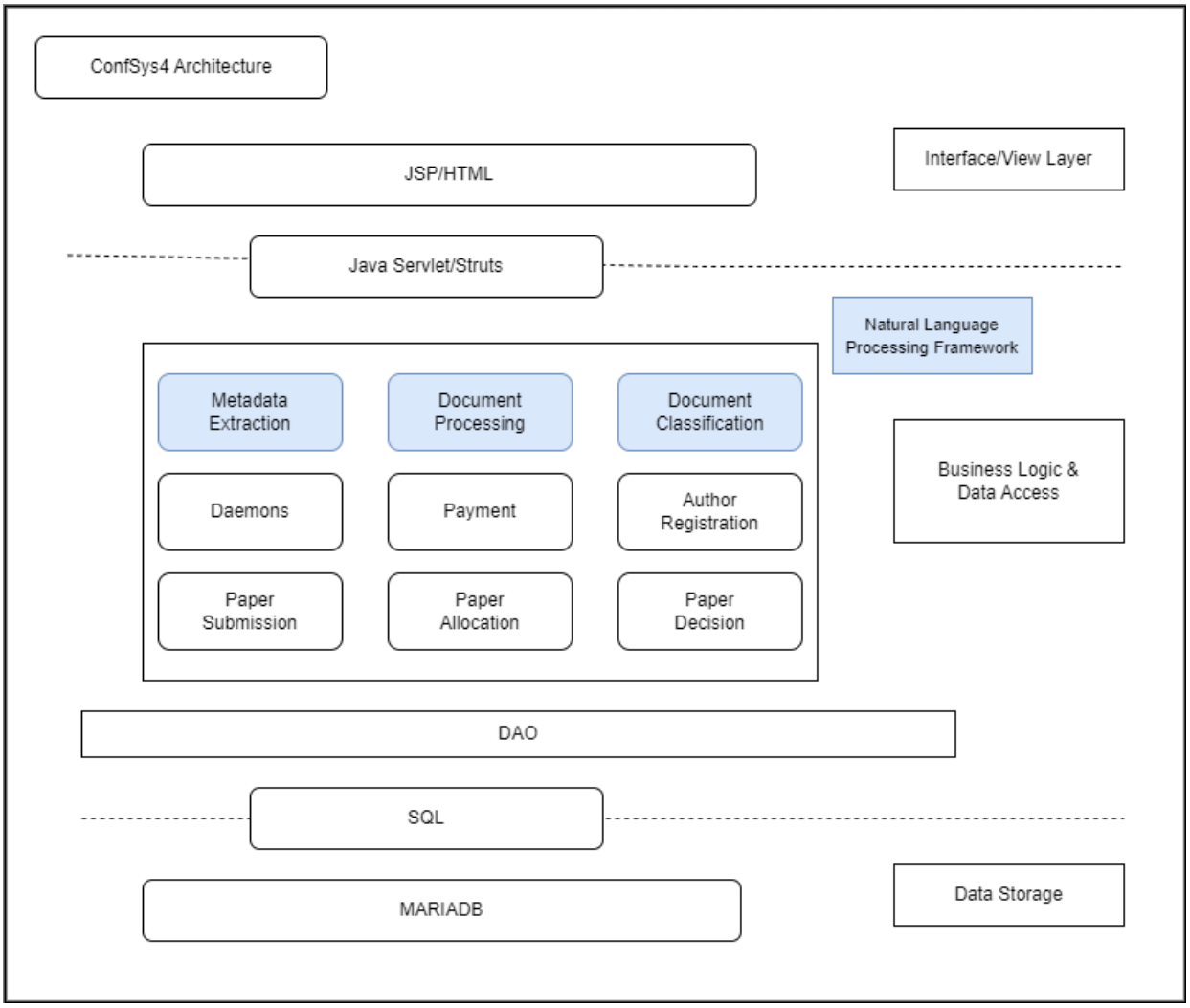

Figure 3.4: ConfSys4 MVC Architecture with newer modules

Chapter 4

ConfSys4 - the improved version

4.1 Overview

This chapter highlights ConfSys4's contributions to improving paper submissions, single/double/triple-blind review, reminder email, author pair identification, PayPal payment interface, and PDF document generation.

4.2 Automatic Meta-data extraction of uploaded PDF file

Paper Submission process is an important feature of conference management systems (CMS). Across CMS system [3, 23–26] described in section 2.2, research papers/abstracts are requested to be submitted for an event. The modes for the paper submission include manual submission over mail, entering paper-related details and uploading documents in a web-based form, and using custom-designed forms and multi-format file upload. The latter provides the flexibility to organizing committees to incorporate changes into submission forms as needed. These modes of submission still require manual intervention by the authors.

Fig 4.1 shows the use case diagram for the paper submission process describing the actors involved, such as author and system, and their associated functionality in the ConfSys system. Important functions such as metadata extraction and display by the system, verification, and metadata update, if required by the author, are implemented into the paper submission process. The metadata

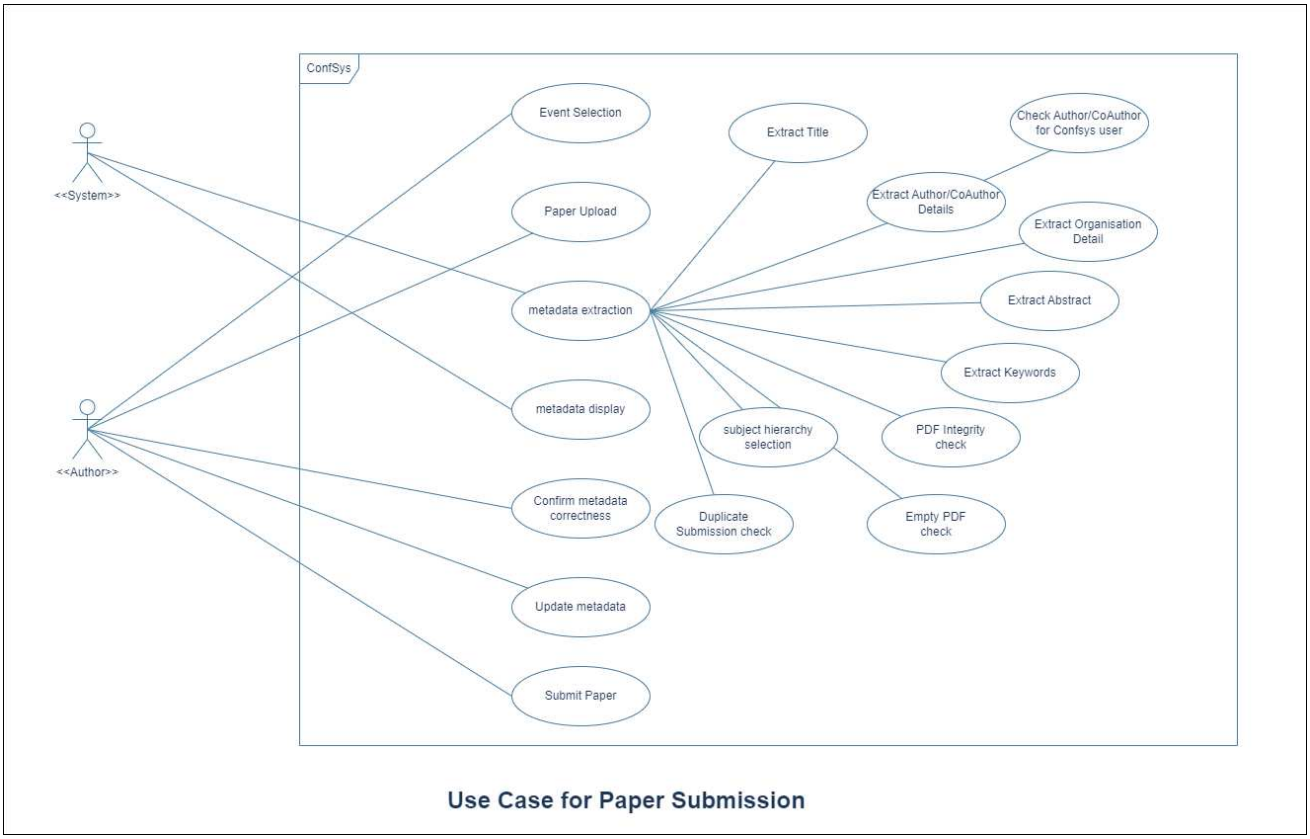

Figure 4.1: Use Case Diagram for Paper Submission Process

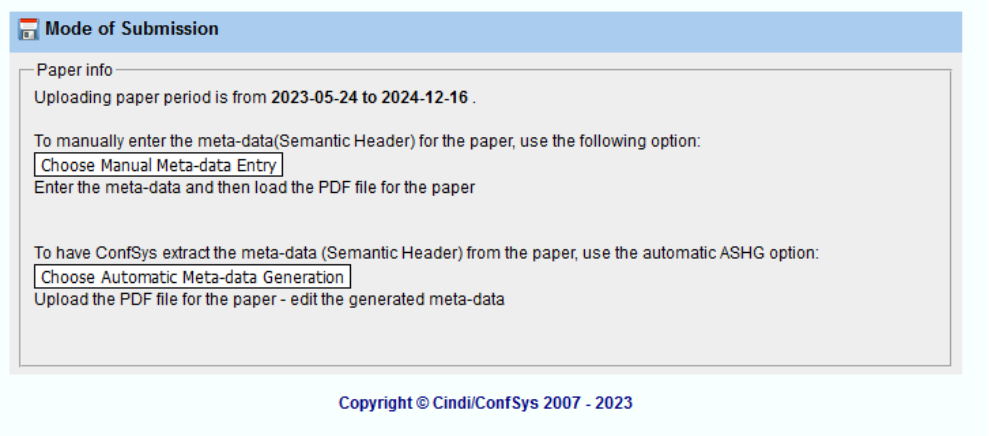

Figure 4.2: Modes of Submission in ConfSys4

extraction functionality is further refined to extract titles, abstracts, keywords, author-related details, subject hierarchy selection, duplicate paper submission check, empty PDF file check, and PDF data content integrity check for submissions.

4.2.1 Metadata Extraction

In ConfSys4, the author is provided with two options for the mode of submission. The manual mode requires authors to add paper-related details and upload PDF documents for submission, whereas the automatic metadata generation mode requires authors to only upload PDF documents for submission, as illustrated in Fig 4.2. In the case of automatic metadata generation mode selection, the author is taken to the submit new paper screen to select the review type and upload the PDF document, as shown in Fig 4.3. Once the author submits a paper, the system automatically processes the uploaded PDF file. It then extracts salient metadata, performs validations, generates logs, and displays extraction status and paper metadata details to the author. The validation check verifies the validity of the PDF document, and if issues are identified, it is reported to the author to rectify the PDF and submit it again. The extraction status log screen, shown in Fig 4.4, provides information about extracted metadata content for the uploaded file to the author.

The author then proceeds to the metadata information page for verification of metadata. The metadata information screen displays extracted details such as title, abstract, keyword, and subject hierarchy selected based on the file's contents, shown in Fig 4.5. In order to identify whether

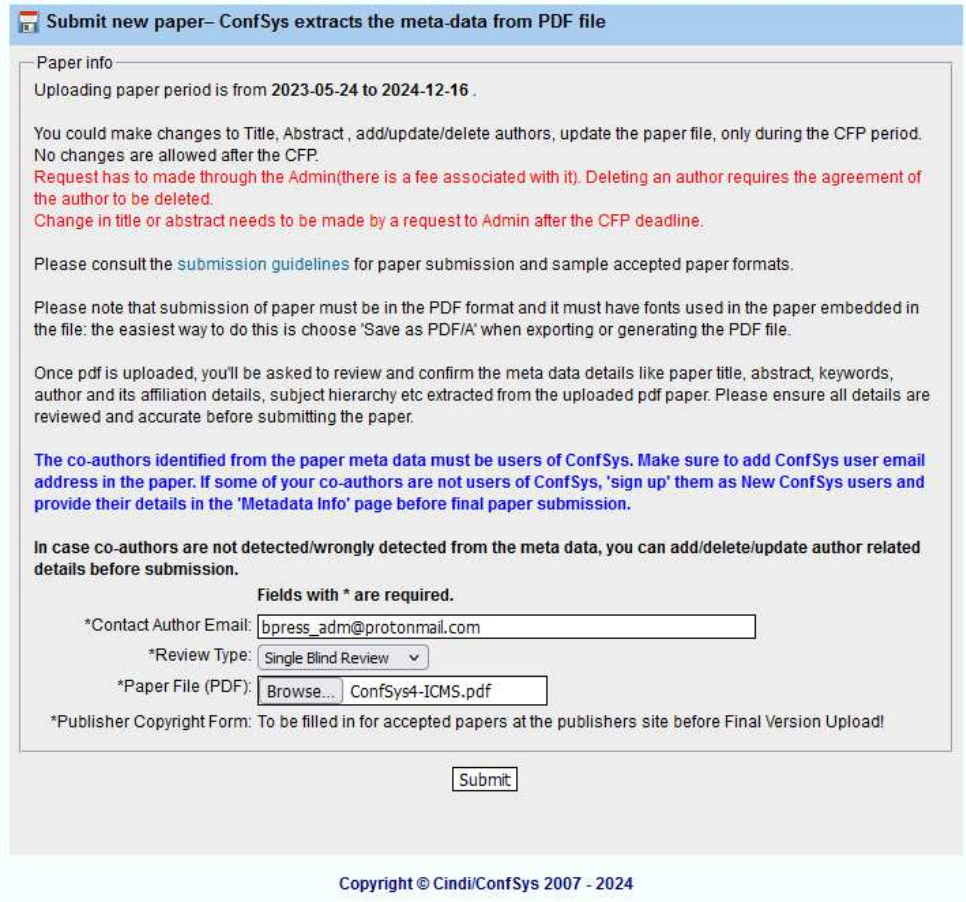

Figure 4.3: Paper Submission in ConfSys4

MetaData Extraction Status:

Metadata Object	Extraction Message
Paper Title	Successfull Extraction. Review and Confirm the extracted metadata contents before submission
Paper Abstract	Successfull Extraction. Review and Confirm the extracted metadata contents before submission
Paper Keyword	Successfull Extraction. Review and Confirm the extracted metadata contents before submission
Author 1	Successfull Extraction - Author Details : Name, Org and Email. Review and Confirm the extracted metadata contents before submission
Author 1	Successfull Extraction - Author Details : Name/Org/Email match to a user already present in ConfSys System
Author 2	Successfull Extraction - Author Details : Name, Org and Email. Review and Confirm the extracted metadata contents before submission
Author 2	Successfull Extraction - Author Details : Name/Org/Email match to a user already present in ConfSys System

Based on status logs generated for metadata objects.You can proceed to the next screen and use one of the following options:
- Review Metadata - Add/Update/Delete metadata extracted and Submit PDF
- Update the PDF Paper based on Extraction Message and Re-Submit PDF
- Use Manual Meta-data Entry Mode and Upload PDF

Continue

Figure 4.4: Extraction Status Log Message

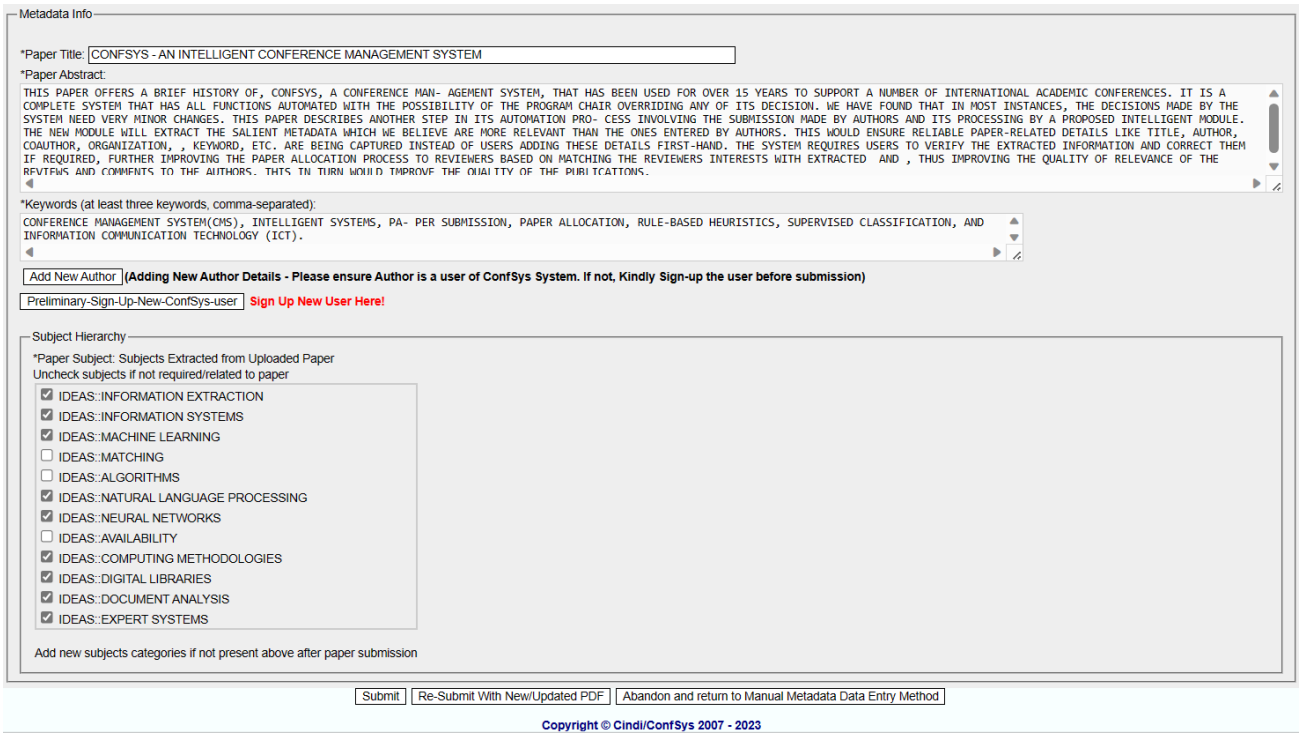

Figure 4.5: Extracted Metadata Details

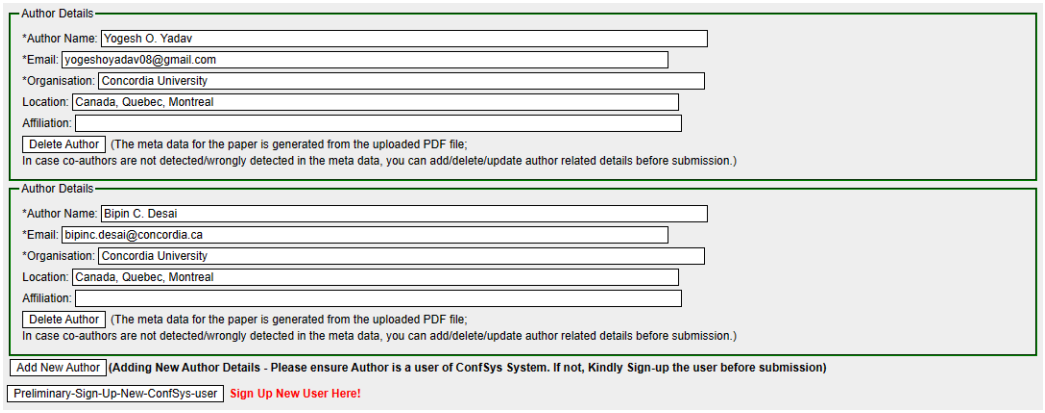

Figure 4.6: Extracted Author Detail Block

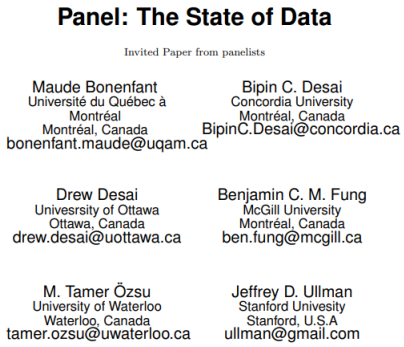

Figure 4.7: Panel Paper Authors Detail

the extracted author metadata is a system user, details such as email, name, and organization are matched and compared with the system user details using the similarity ratio score described in Section 3.3.3. If the extracted author details match, the paper is associated with the matched user details present in the system. If not matched, the submitting author is asked to do a preliminary sign-up for the new users before submitting the paper, as shown in Fig 4.6. The author is required to review all the metadata items and, if required, update its contents.

4.2.2 Handling variation between extracted Author data and ConfSys database

The author-matching process uses a priority-based approach and similarity ratio score described in Section 3.3.3 to compare extracted author metadata with the system user details. The email is given priority and, if present in the PDF author metadata, is used for author-matching. If the email is not present or does not match the system user emails, a combination of the author name and organization is used for author-matching. The similarity threshold is set to 0.7. If the similarity ratio score for the author name and organization present in PDF author metadata and system is greater than or equal to 0.7, it is considered to be similar.

Fig 4.7 displays the panel paper submitted to the ConfSys4 system with one of the authors, ”Jeffrey D. Ullman,” to test system behavior in handling variations in the author name. The author ”Jeffrey D. Ullman” is a user of the system, having signed up as ”Jeffrey David Ullman”. Once the panel paper is submitted, the system first compares the email present in the PDF with system

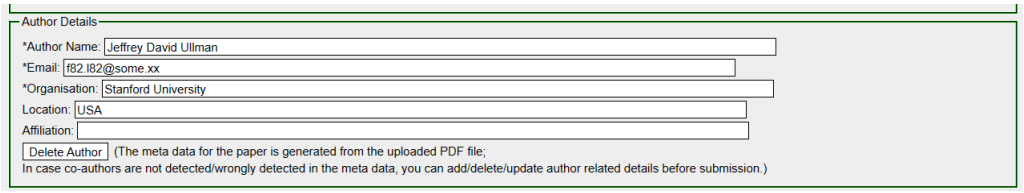

Figure 4.8: ConfSys System User Details for Jeffrey D. Ullman in Panel Paper

user emails. If the email match is not found, the similarity ratio score is used for matching and comparing the author's name and organization. The PDF's author "Jeffrey D. Ullman" is matched to the system user detail "Jeffrey David Ullman". The submitted paper is then associated with the matched system user details. The system user details are then displayed to the submitting user for verification, as shown in Fig 4.8. A typical user can have multiple email addresses and use separate emails to create a ConfSys system account and submit a paper to a conference event. Thus, the author matching using name and organization restricts multiple account creation for the same user with different email addresses. The submitting author is required to verify the metadata details and, if required, update the author details and perform preliminary signup of the paper authors who are not present in the system before submitting the paper.

4.2.3 Duplicate Paper, Empty PDF and PDF Contents Data Integrity Check

The ConfSys system checks extracted metadata contents such as paper title, abstract, and keywords for the paper in submission with the already submitted paper to restrict multiple submissions of the same paper for an ongoing event. The duplicate submission check is performed using the similarity ratio score described in Section 3.3.3 with the similarity threshold of 0.7. The similarity ratio score for paper title is given priority and verified first with the already submitted paper titles. If the paper title is not matched, the paper abstract and keywords are verified for the submitted PDF and system paper details. The author is allowed to submit a new version for the same paper but is not allowed to create a new submission entry and upload the same PDF of the paper into the ConfSys system, as shown in Fig 4.9.

In addition, the system also checks the contents and size of uploaded PDF documents. In case the author submits an empty PDF file, the system processes the document and displays a rejection

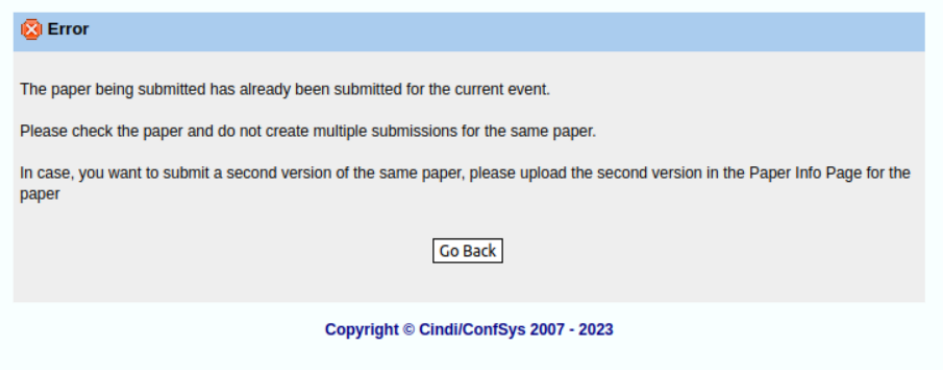

Figure 4.9: Duplicate Paper Submission Message

Figure 4.10: Paper Submission with Empty PDF

message shown in Fig 4.10 to notify the user to submit a valid PDF file for the paper submission. In cases, if the PDF file in submission is generated using images or tabular format, or unsupported encoding format for paper metadata and author metadata contents, the system displays a rejection message shown in Fig 4.11 to notify the user to submit a PDF file with metadata details added in a text-based encoding format. Fig 4.12 shows pre-processed XML data generated for an invalid PDF document submitted to the system. The PDF contents are present in XML node ”figure” instead of node ”textbox” as required for extraction described in Section 3.2.

4.2.4 Improved Paper Submission With ASHG

ConfSys4 system has improved the paper submission process to incorporate metadata extraction and subject hierarchy classification for the submitted paper. The following is the summary of new

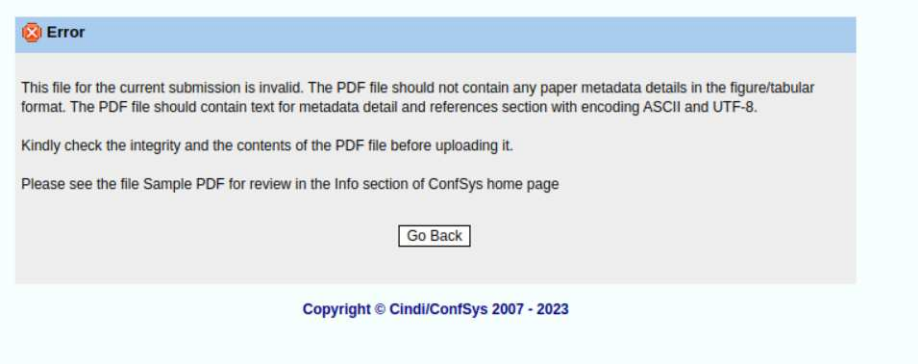

Figure 4.11: Paper Submission with Invalid PDF Contents

Figure 4.12: XMl data for the Invalid PDF File

features added to the paper submission process :

- Metadata extraction from the metadata and author section present in the PDF document
- Subject hierarchy classification based on the contents present in the PDF document
- Status logs to provide information about extracted metadata to submitting author
- Extracted metadata review and updates by submitting author
- Preliminary sign-up of new users by submitting authors
- Submission modes such as submit with extracted metadata, or submit with manual data entry for paper-related details are available to submitting authors
- Duplicate paper submission check to restrict multiple submissions of the same paper
- Empty PDF file check to restrict invalid submissions
- PDF file data integrity check to restrict PDF submissions with paper metadata details in images/tabular or unsupported encoding format

4.3 Author Details in PDF file for Single/Double/Triple Blind Review

ConfSys4 improves the single/double-blind review process by incorporating document content verification for author details in the author metadata and references section of the uploaded PDF file. The author selects the appropriate review type (single/double) during paper submission. Based on the selected review type, the uploaded PDF file is processed to check and verify whether the file should include author details in its metadata or reference sections of the PDF. The single-blind is commonly used and is less restrictive to anonymity allowing reviewers to know the identity of the author but not vice versa. In double-blind, the reviewers and the authors do not know the identity of each other, useful when one or more authors of the paper belong to the program committee members. Triple-blind review selection is only available to roles such as Admin, General Chair, and Program Chair. In a triple-blind review, the reviewer does not know the author's identity, and the Program Chair and other program committee members do not know the reviewer's identity.

Useful when one or more paper authors belong to the Track Chair, Program Chair, or General Chair role. The single/double-blind review selection is available to all users for paper submission. The paper submitted for single-blind review is verified by the system to ensure authors of the paper do not belong to organizing committee members. If one or more paper authors are identified as organizing committee members, the system requests the author to modify the PDF by removing author metadata and reference citations of paper authors and submit the PDF again as double-blind. In case of double-blind submission, if the authors added by the submitting author belong to the Track Chair, Program Chair, and General Chair, the system automatically updates the review type for the paper to triple-blind.

The review selection in the ConfSys4 system is similar to the previous versions [3] to ensure anonymity in the paper review process. The Admin and General Chair users are only allowed to submit papers for other authors and update the review selection for the submitted paper. Triple-blinded papers are only accessible to users with roles such as Admin and General Chair for allocating reviewers, making paper decisions, and viewing paper details. The users with roles such as Program Chair, Track Chair, or Program Committee members cannot access triple-blinded papers. In the case of double-blinded papers, users with roles such as Admin, General Chair, and Program Chair are allowed to allocate reviewers, make paper decisions, and view paper details. The users with the role of Track Chair are only allowed to view paper details for double-blinded papers. ConfSys4 system processes the submitted PDF to incorporate checks for single/double/triple-blind review based on the review type selection by the author during the paper submission process.

4.3.1 Single Blind Review Selection

The single-blind review selection allows authors to keep author-related details in the metadata section and reference citations of the paper authors in the reference section of the PDF document. If the submitting author selects a single-blind review for the paper submission, it could lead to the following scenarios during the submission :

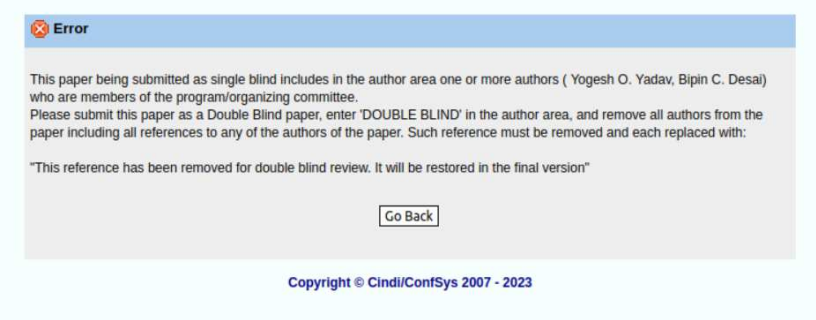

Figure 4.13: Paper Submission with One or more Paper Authors associated with Organizing Committee

One or more authors of Paper belong to Program Committee members

In case one or more authors of the paper belong to the organizing committee and the submitting author selects single-blind review during paper submission, the uploaded PDF file is processed, and the extracted author details are compared and matched to the system details of program committee members using similarity ratio score ratio described in Section 3.3.3. If the author match is found with a similarity ratio score greater than the threshold similarity limit (0.7), the system rejects the paper, as shown in Fig 4.13, and requests the author to update the PDF document to exclude author details in the author metadata section and any reference citations associated with the paper authors who are members of organizing committee and resubmit the paper as double-blind.

No authors of Paper belong to Program Committee members

In case no authors of the paper belong to the organizing committee, the author's single-blind submission of the paper is checked and verified for the submitting user details. It is required that submitting author details should be present in the author metadata section of the PDF. The only exception is given to the Admin and General Chair role to be able to submit for other authors. If submitting author details are absent in the author metadata section, the uploaded PDF file is rejected as shown in Fig 4.14. On the contrary, if submitting author details are found in the author metadata section, the uploaded document is processed, metadata details related to the paper and authors are extracted, and the author details are compared and matched to the system users using the similarity ratio score ratio described in Section 3.3.3. If extracted author details are not present

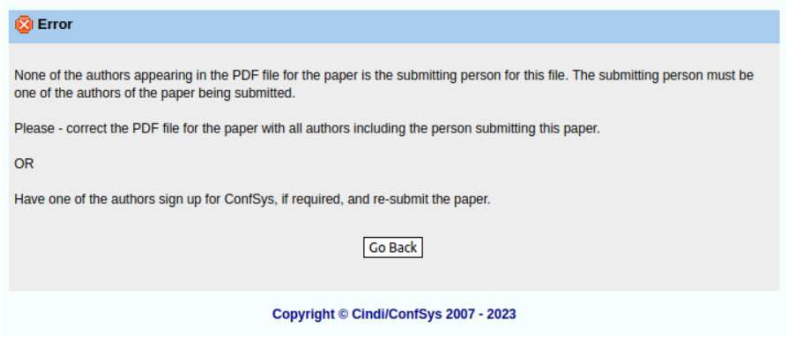

Figure 4.14: Paper Submission with submitting author details not present in Author Metadata Section of PDF

in the system, the submitting author is required to perform a preliminary sign-up for the new users before submitting the paper.

4.3.2 Double Bind Review Selection

The double-blind review selection requires excluding author details and references from the author metadata and references sections for the paper authors who are members of the organizing committee. If the author submits the PDF as double-blind and the uploaded PDF contains one or more author details, the system rejects the submission and requests the author to resubmit the paper after removing author details, as shown in Fig 4.15. If the PDF submission with no author details in the author metadata section is successful, the author is asked to review the extracted metadata and add author details for the paper's authors. The submitting author is considered a contact author (paper author) by default. In scenarios where users having Admin/General Chair roles submit the paper as double-blind, they are not considered as paper authors. The submitting author's details are not allowed to be updated or deleted during the verification of metadata contents by the author, as shown in Fig 4.16. The authors to be added are required to be a user of the system beforehand. The author submits the paper after reviewing the extracted metadata and adding all the authors of the paper. Once submitted, the author's details are compared with the author list generated from the author references sections. If the paper author name matches any author in the author list using the similarity ratio score described in Section 3.3.3, the submission is rejected, and the submitting author is requested to remove the reference citation from the PDF and resubmit the paper as shown

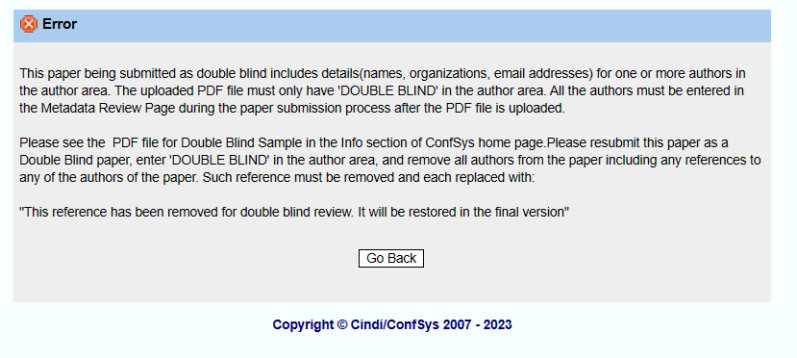

Figure 4.15: Paper Submission for Double Bind with Author Details

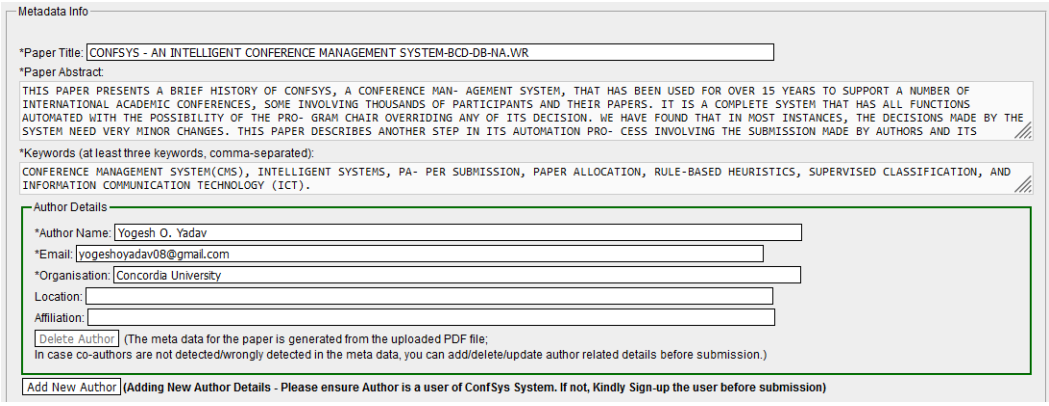

Figure 4.16: Double Bind with Submitting Author Details

in Fig 4.17.

4.3.3 Triple Bind Review Selection

Triple-blind selection is only available to Admin/General Chair/Program Chair roles. In case of double-blind PDF submission, one or more authors added in the metadata review page belong to the Track Chair, Program Chair, or General Chair role, the system compares and matches the author email with the system user email associated with Admin/General Chair/Program Chair/Track Chair role; If an author email match is found, the review selection is automatically changed to triple-blind for submission as shown in Fig 4.18.

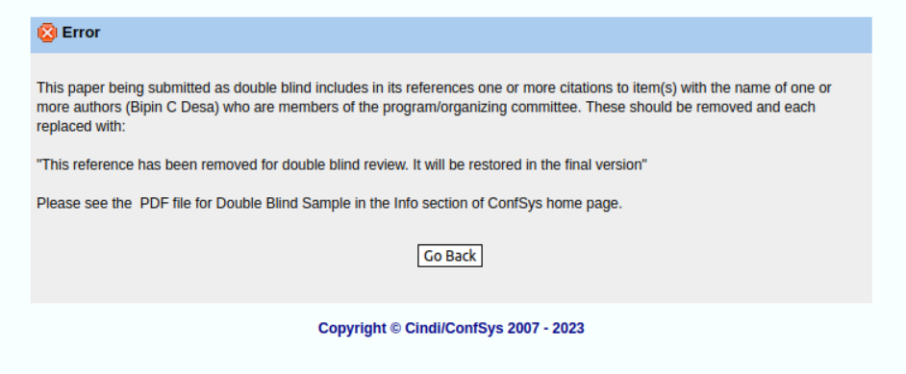

Figure 4.17: Paper Submission for Double Bind with Author Citations

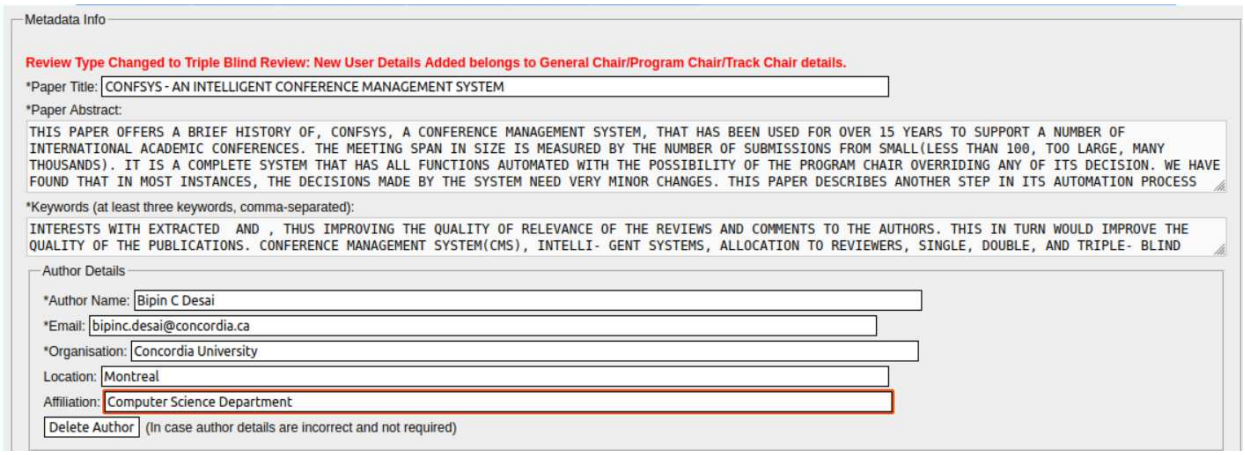

Figure 4.18: Paper Submission with General Chair/Program Chair/Track Chair as Author

4.4 Manual Mode Submission

The manual mode submission option in ConfSys4, as shown in Fig 4.2, allows authors to manually enter paper-related details and upload a PDF document for submitting the paper. The submission process is similar to ConfSys3.5, except the PDF document is processed to verify author metadata and reference section present in the PDF for single/double/triple-blind review. The author-entered paper-related details are also verified and compared to already-submitted papers to enforce duplicate submission checks. The verification for single/double/triple-blind, duplication check and invalid PDFs are similar to the automatic metadata generation mode described in Section 4.2 and Subsections 4.3.1, 4.3.2, and 4.3.3. The author is notified with an appropriate rejection message if any of the verification checks fail.

4.5 Automatic Reminder Emails for Program Committee Member for Topic of Interest

Paper allocation and review is another essential process in conference management systems to efficiently assign papers among the organizing committee members based on their expertise and topics associated with the submitted paper. It requires submitted papers tagged to appropriate subject classifications and program committee members with topics of expertise already selected for consistent paper allocation among the members. Similar to the previous ConfSys [3], ConfSys4 tries to find the best match based on the paper's subject hierarchy and the program committee member's topic of expertise before resorting to a random match. Automatic tagging to subject classifications based on uploaded PDF content is handled by the extractor module during the paper submission process, as shown in Fig 4.5. On the other hand, program committee members require persuasion to update their topic of interest. ConfSys4 incorporates a reminder email daemon routine that executes regularly and reminds program committee members to update their topic of interest if not already updated until the auction period is reached. The reminder email is shown in Fig 4.19.

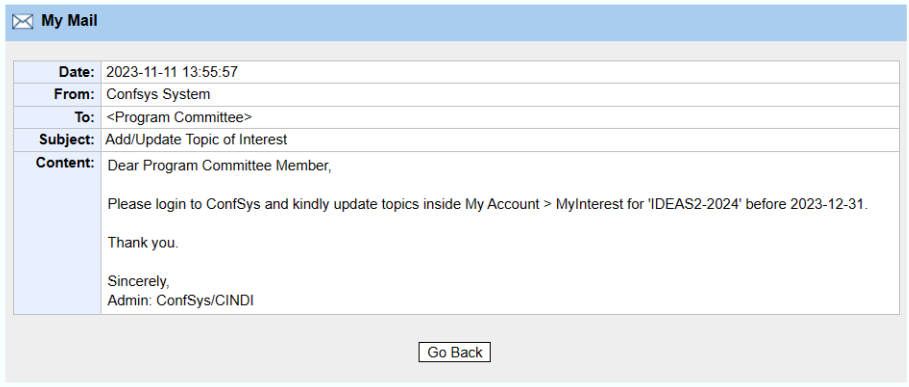

Figure 4.19: Automatic Reminder Emails

4.6 Improved Author Pair Identification

Related author pair identification process from the DBLP datasets [5] is improved in ConfSys4 compared to ConfSys3.5. The ConfSys4 system uses a daemon routine for DBLP data processing to execute regularly on the last day of every month or allows the Admin role user to execute all daemon routines via the ConfSys web interface. Furthermore, it downloads and stores the DBLP MD5 (Message Digest Algorithm 5) signature file from the official DBLP site into the system for processing. MD5 is used to create digital signatures and is often used to verify the integrity of files. It produces a 128-bit hash value (32 hexadecimal characters) from any arbitrary amount of input data. For each processing of the DBLP dataset, the MD5 signature is downloaded and compared to the previously processed DBLP MD5 signature. If not matched, the DBLP file is downloaded for further processing, or if the MD5 signature is matched, the process is skipped to save time due to redundant processing. Once downloaded and unzipped, authors are consolidated together, and an authors list is generated for each paper record in DBLP. ConfSys4 then iteratively compares and matches each author present in the author list with system user details. If the match is found, the remaining authors in the list are compared with the system users. If matched, the author pair is created and stored in the system with relation as ”DBLP Coauthor”. The author match process in ConfSys3.5 [3] used an exact matching technique based on the author’s name and could miss identifying potential pairs with minor differences in their names. In ConfSys4, the author match process is improved by using the similarity ratio score described in Section 3.3.3 with a similarity

AUTOMATIC 2D TO STEREOSCOPIC VIDEO CONVERSION FOR 3D TVS

Xichen Zhou†, Bipin C. Desai, Charalambos Poullis†

Immersive and Creative Technologies Lab†
Department of Computer Science and Software Engineering
Concordia University

Figure 4.20: Paper Metadata for DBLP Paper

```
MariaDB [ConfSys1]> select * from dblp where authors like '%Xichen Zhou%';
+--------+-----------------------------------------------------+
| dblpid | authors                                              |
+--------+-----------------------------------------------------+
| 549662 | |Bipin C. Desai|Charalambos Poullis|Xichen Zhou| |
+--------+-----------------------------------------------------+
1 row in set (1.228 sec)

MariaDB [ConfSys1]> select relationid,u1.firstname,u1.lastname,u2.firstname,u2.lastname,relation from user_relation ur
ur.userid2 = u2.userid where ur.userid1=69 and ur.userid2=2076 and relationid=839775  limit 10;
+------------+-----------+----------+-----------+----------+---------------+
| relationid | firstname | lastname | firstname | lastname | relation      |
+------------+-----------+----------+-----------+----------+---------------+
|     839775 | Bipin     | Desai    | Xichen    | Zhou     | DBLP coauthor |
+------------+-----------+----------+-----------+----------+---------------+
1 row in set (0.000 sec)
```

Figure 4.21: DBLP Record and Coauthor Paid Identification

threshold of 0.7 to improve author name matching with minor variations in their names.

Fig 4.20 shows author metadata for the paper ”AUTOMATIC 2D TO STEREOSCOPIC VIDEO CONVERSION FOR 3D TVS” published by ”Bipin C Desai” and ”Xichen Zhou”. The paper record was processed during the DBLP dataset processing, and the author list was consolidated, as shown in Fig 4.21. The coauthor pairs were extracted from the author list of the paper and matched with the system user details. The coauthor pair for ”Bipin C Desai” and ”Xichen Zhou” is identified and matched with the system user details, and the ”DBLP Coauthor” relation record is added to the system, as shown in Fig 4.21.

4.7 Update to PayPal Payment Interface

The ConfSys3.5 [3] system supports payment integration for user registration to events with widely used third-party payment applications such as PayPal and Moneris. In ConfSys4, the payment interface for the PayPal payment mode is improved by using PayPal’s standard checkout Rest

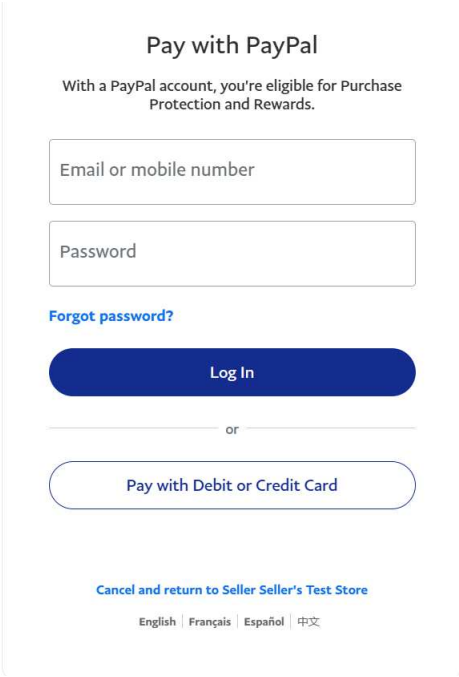

Figure 4.22: Improvement in Paypal Payment Interface

API for payments. The standard payments allow options such as Pay with PayPal account and Pay with debit/credit card details to ensure easy, safe, secure, and reliable payments, as shown in Fig 4.22. The payment response message from PayPal is used to update the registration status for the user for the event. If the payment is successful and confirmed from the response message received, the details are recorded and stored in the system, as shown in Fig 4.23. The PayPal developer account is set up to get the sandbox environment credentials. The sandbox credentials, such as Client ID and Client Secret, are used to authenticate and authorize the rest API calls. Sandbox test card details for debit/credit are used to test the standard payment integration to the Paypal payment mode.

4.8 PDF Document Generation

In ConfSys4, the download PDF option is available for the registered users to download their invoice and payment receipts after registration to the event and successful payment, respectively. Thus allowing attendees to conveniently access and retain their invoices and payment receipts, ensuring a streamlined and organized registration process. The invoice/payment information is dynamically

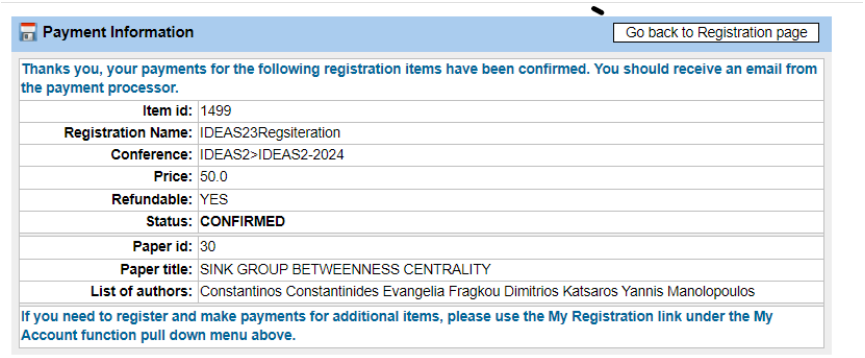

Figure 4.23: Payment Success

overlayed on the system-defined letterhead for PDF generation. The lower half in Fig 4.24 shows the download button present to generate payment receipts after successful registration and payment. Once the user clicks on the button, the PDF for the payment receipt is downloaded, as shown in Fig 4.25. A similar button is added to the web interface for downloading the preliminary/final program in PDF format, as shown in Fig 4.26. This allows the Program Chair to download and share the PDF version of the program with the registered users and organizing committee members of the event. This, in turn, makes it easier for participants to access and review the program offline, facilitating better engagement and planning for the event. The downloaded PDF document for the program is shown in Fig 4.27.

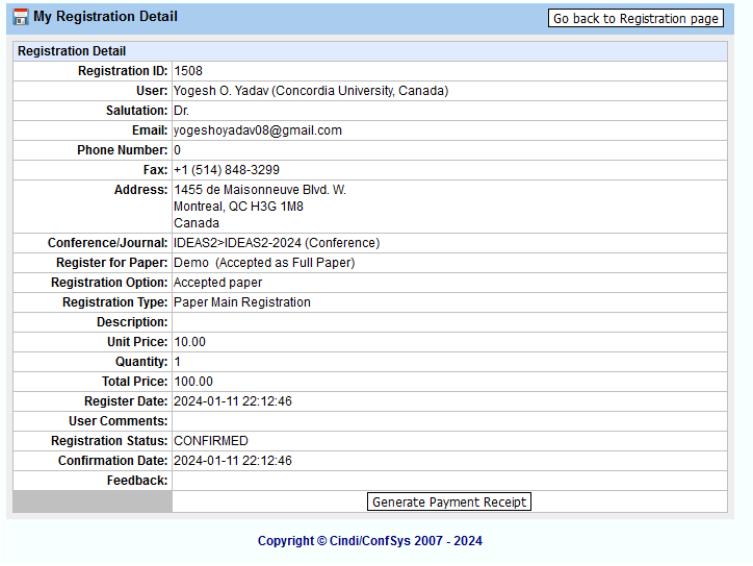

Figure 4.24: Generate Payment Receipt Button for Downloading Payment Receipt

Figure 4.25: Downloaded PDF for Payment Receipt

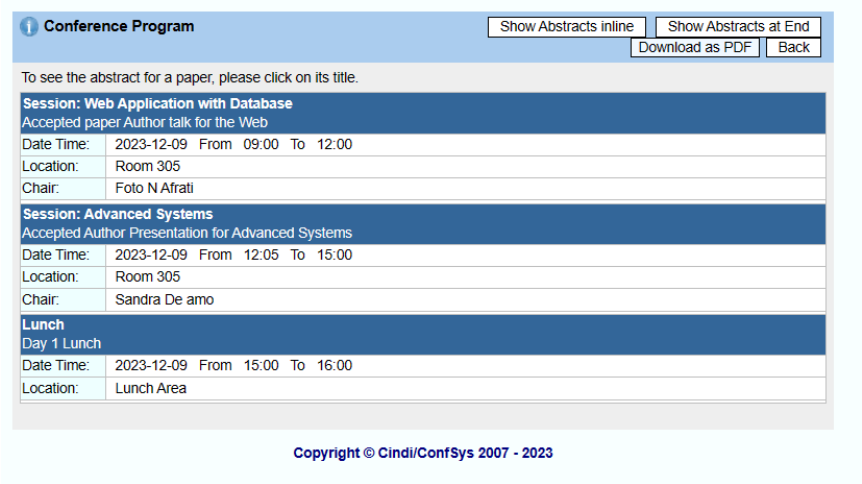

Figure 4.26: Download PDF button for Program

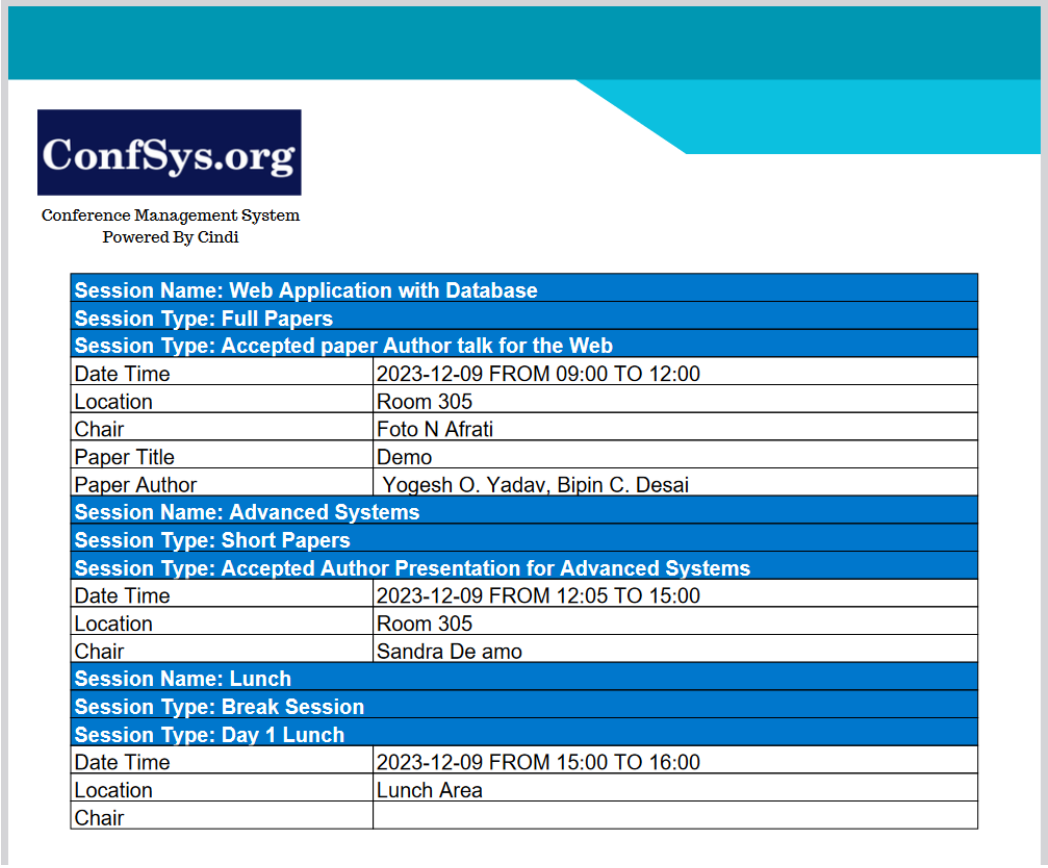

Figure 4.27: Downloaded PDF for Program

Chapter 5

System Implementation

5.1 Overview

This chapter describes the implementation details for the metadata extractor system ASHG 2. It provides information related to the architecture, and all the individual tasks involved in Automatic Semantic Header Generation (ASHG 2) for an information resource.

5.2 Automatic Semantic Header Generator (ASHG 2)

5.2.1 Architecture

The Metadata Extractor (ASHG 2) system accepts a single PDF file as input during the paper submission process. The PDF file is passed to the document processing step, where it is converted to a machine-readable XML format file. The conversion of the input PDF to XML provides additional features such as font type, font size, geometric position (bounding box), relative position (X, Y), and textual-based features. The XML formatted file is then passed to the pre-processing task where all leaf nodes ("text") contents are consolidated, passed to their immediate parent node ("textbox"), and deleted, as previously described in Section 3.2 and shown in Figs 3.1 and 3.2. This pre-processing task reduces the size of the XML in comparison to the raw converted XML file. The XML traversal becomes easier and faster in the further steps for the extractor system. The pre-processed XML file is then inputted to the ASHG 2 extractor's metadata object and subject hierarchy classification

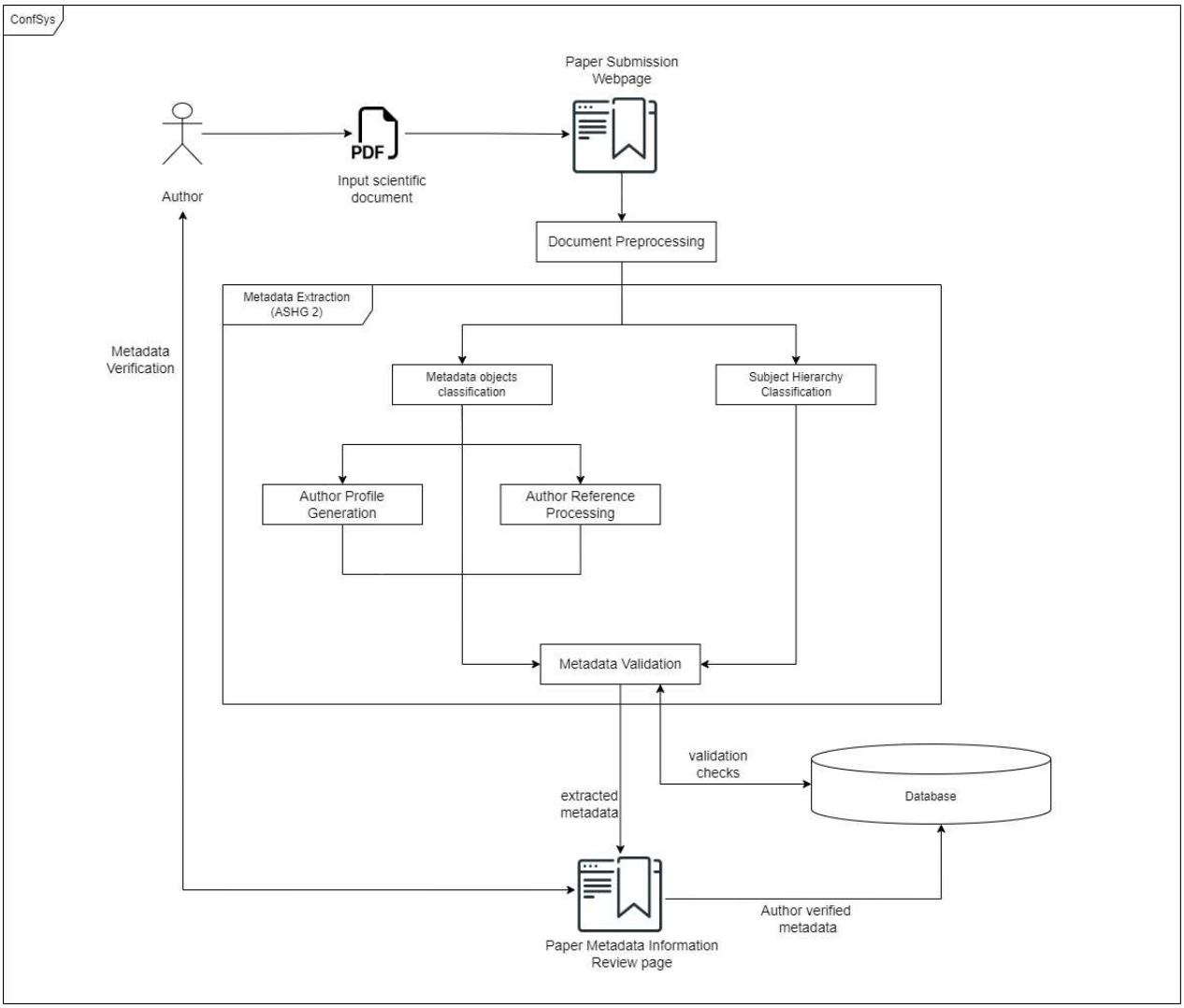

Figure 5.1: Architecture for Metadata Extractor (ASHG 2) in ConfSys4

tasks. The metadata object classification task classifies input nodes to output class labels such as title, abstract, keyword, author, and other. The XML nodes with class labels ”author” and ”other” are further processed to generate author profiles and author reference lists. The subject hierarchy classification task generates an implicit list of subject headings based on the contents of the PDF. Furthermore, the classification task outputs are passed to the validation step to verify the metadata quality. The output is then consolidated to generate a semantic Header to share with the user for review. The user verifies and, if required, updates the extracted metadata before paper submission. After successful submission, extracted metadata is stored and committed to the system database. The individual steps of the metadata extraction are shown in the architecture diagram in Fig. 5.1.

5.2.2 Metadata Objects Classification

This is the most critical step in the algorithm, where the metadata objects such as title, abstract, keywords, author profile generation, subject hierarchy selection, and author reference list are extracted from the modified XML file. The classification task is further divided into separate sub-tasks, as shown in Table 5.1. The table clearly outlines the extraction approach for each of its sub-tasks. The library scikit-learn [27] is used for training and testing the classifier models such as Conditional Random Field (CRF) [41] and Naive Bayes (NB) [42] for the classification task. Named Entity Recognition for author profile generation and author reference extraction is implemented using natural language processing frameworks such as Natural Language Tool Kit (NLTK) [28], Spacy [29], BERT-NER model [32] and regular expression [33]. The similarity ratio score described in Section 3.3.3 is used for comparing the similarity of two strings in the author profile generation task. In addition, the system user details such as name, email, organization, department, and address are used as Gazetteer data in the author profile generation task. The execution details implemented in Python language for the ASHG2 extractor are described in Table 5.2. Model storage details for the classification and named entity detection model are added in Appendix Section A.3.

Task	Libraries Used	Functionality/Approach	Output/Response
Metadata objects classification	rules, scikit-learn	Rule-based logic, Discriminative Conditional Random Field (CRF) classifier	XML data with class labels (title, abstract, keyword, author and other)
Subject hierarchy classification	rules, scikit-learn	Probabilistic Naive Bayes (NB) classifier	List of subject headings
Author profile generation	Hugging Face supported BERT Models, NLTK, Spacy, Levenshtein	Named Entity Recognition, Gazetteer list, Regular expression, Levenshtein ratio, Reading order logic	Nested dictionary object containing author and its details
Author references processing	Hugging Face supported BERT Models, NLTK, Spacy, Levenshtein	Named Entity Recognition, Gazetteer list, Regular expression, Levenshtein ratio, Reading order logic	List of reference author names

Table 5.1: Metadata Extractor Module Tasks

Scritps	Details
Metadata_Extractor.py	Main execution script for calling methods such as pdf_to_xml.py, preprocess_xml_data.py, and extractor.py. Accepts PDF document as inputs
pdf_to_xml.py	Uses PDFMiner library for conversion of PDF to raw XML formatted documents.
preprocess_xml_data.py	Uses LXML library for XML processing. Includes pseudo code for processing raw XML to pre-processed XML document described in Section 3.2.
extractor.py	Uses Scikit-learn library for importing classifier models such as Condition Random Fields (CRF) and Naive Bayes(NB). Includes pseudo-code for algorithms 2 and 3 described in Section 5.2. Accepts pre-processed XML as inputs and returns extracted metadata objects.
named_entity_detection.py	Uses NER libraries such as Natural Language Tool Kit (NLTK), Spacy, Hugging Face, Transformers (BERT), Levenshtein. Includes pseudo-code for detection of named entities such as name, email, organization, and location. Accepts text as input from extractor.py and returns named entity labels associated with text tokens

Table 5.2: ConfSys4 Python Module Implementation Details

Title Extraction

Most scientific documents have a similar format for representing the title of the paper in their layout structure. For example, font type is bold, font size is maximum and located at the top section of page 1 of the paper, in a way, making it predictable for developing rule-based patterns for consistent extraction of paper titles from the documents. In title extraction, rule-based logic is implemented based on layout attributes such as font type and font size present in the input XML data. The input XML Page 1 node contents are searched for the highest and second-highest font size nodes. Node data contents are consolidated for both font sizes and stored into string objects. The Max size font object is then passed to the title validation check. The validation checks include verifying whether special characters were present, the total word count to have at least 4 and at most 50 words, and the character count to have more than 20 characters. If passed, extracted content is considered as title, otherwise, a similar validation check is performed for the second max font size string object. The title extraction using rule-based heuristics is computationally less expensive with a faster response time.

Other metadata objects

Metadata objects such as abstracts, keywords, author details, author references, and introductions are extracted using the machine learning model Condition Random Field (CRF) [43–45] available in Scikit-learn library [27]. The Scikit-learn library described in Section 3.3.1 provides tools for data pre-processing, model selection, training, testing, and evaluation. ConfSys4 uses the scikit-learn library for model selection (CRF), training/testing, evaluation, and classification of metadata objects. In order to extract these metadata objects; it is essential to give importance to the contextual information of the sentence, such as sentence structures and neighboring words. The task of predicting labels for the sequence of sentences interdependent rather than a single independent sentence falls in the category of sequence classification problems.

In Sequence classification models [46], for each input sequence x_i in X, we assign a label y_i so that output sequence Y has the same length as the input sequence X. The sequence classification models are further classified into two categories: generative and discriminative models. Generative

models such as the Hidden Markov Model (HMM) [47] capture the joint distribution, $p(y, x)$, considering both the input sequence X and the corresponding labels Y. HMMs compute the probability for a sequence of observable events X, involving hidden states and observable outputs. However, HMMs have the limitation that it is difficult to model arbitrary, dependent features of the input sequence. On the other hand, Discriminative models such as CRF [41] capture the conditional distribution $p(y|x)$. CRFs consider the input sequence X and aim to maximize the probability distribution of the output sequence Y given input sequence X, emphasizing the relationship between input features and output labels without the explicit need to model $p(x)$ as they are not needed for classification. Overlapping, non-independent features are possible to model with CRF classifiers. As a special case, linear chain CRFs can be thought of as the undirected graphical model version of HMMs. In order to maximize the conditional probability $P(Y|X)$, given an input sequence X (XML data) and output labels Y (abstract, keyword, author, and other), the following formulae is provided:

$$P(Y|X) = \frac{1}{Z(X)} \exp\left(\sum_{i=1}^{N}\sum_{k=1}^{K} \lambda_k \cdot f_k(y_{i-1}, y_i, xi)\right) \tag{2}$$

where

- $Z(X)$: is the normalization factor that ensures the distribution sums to 1 over all possible output sequences
- λ_k are the learned model parameters
- $f_k(y_{i-1}, y_i, xi)$ is the feature functions for the classifier, including current output state label y_i, the previous output state label y_{i-1}, and the current input features derived for an observed event x_i

The feature functions $f_k(y_{i-1}, y_i, xi)$ defined encodes the dependencies between current input features and output labels. It is an important component in CRF, thus defining feature functions is crucial to capture dependencies between input XML node data elements attributes and the output labels for metadata objects. Table 5.2 shows a list of Features in the Feature Function for the CRF

Category	Features
node-based	token_count(node.elem)
lexical-based	is_NER_present(node.elem)
rule-based	is_token_count_more_15_words(node.elem)
rule-based	is_comma_separated_words_present(node.elem)
rule-based	is_all_alphabetic_characters(node.elem)
rule-based	is_all_numeric_characters(node.elem)
rule-based	is_alphanumeric_characters(node.elem)
rule-based	is_metadata_object_phares(node.elem)
font-based	is_fontType_bold(node['fontType'])
font-based	is_curr_node_fontSize_same(curr_node['fontSize'],prev_node['fontSize'])
font-based	is_curr_node_fontSize_smaller(curr_node['fontSize'],prev_node['fontSize'])
font-based	is_curr_node_fontSize_greater(curr_node['fontSize'],prev_node['fontSize'])
font-based	node['fontType']
font-based	node['fontSize']
font-based	is_maxFontSize(node['fontSize'])
position based	node['bbox']
position based	node_relative_postion(node['bbox'])

Table 5.3: CRF Feature Function

classifier. It includes node-based, layout-based, lexical-based, and position-based features along with output state class labels for the current and previous observed events.

The steps involved in CRF model training, testing, evaluation, and classification are explained in the following bullet pointers and the pseudo-code is provided in Algorithms 1 and 2.

- Preprocessing and Data annotation- The labeled metadata dataset containing the paper title, abstract, keywords, subject hierarchy, author details, and the associated raw PDF documents are considered. The PDF document is converted to intermediate XML documents as mentioned in the algorithm in the document preprocessing step. Each XML node is tagged with output labels such as title, abstract, keyword, author, and other based on node contents and labeled metadata. This step is shown in Algorithm 1 from lines 1 to 9.

- Feature Extraction - Iterate over the XML node and pass the node contents and attributes to the feature function to generate features. Associate extracted features and metadata labels for each node for model training and testing. This step is shown in Algorithm 1 from lines 10 to 23.

- Model Training - A liner chain CRF model is initialized with parameters such as algorithm, L1

and L2 regularization parameters, max-Iterations, and all-Possible-Transitions. The classifier model is then trained on the training dataset containing features and metadata labels for the input node from the feature extraction step. This step is shown in Algorithm 1 from lines 24 to 27.

- Model Testing and Performance Evaluation - The model is tested on the testing dataset. The ground truth and model results are compared using the similarity ratio score described in Section 3.3.3. The accuracy of the model output on testing data is evaluated. This step is shown in Algorithm 1 from lines 28 to 29.

- Parameter Adjustment - The classifier model is further fine-tuned to maximize the accuracy score. Regularization parameters L1 and L2 are tweaked to prevent overfitting and bias-variance trade-offs. Feature selection to reduce dimensionality and improve model discrimination and prediction accuracy. This step is shown in Algorithm 1 on line 30.

- Storing the Model - Once model parameters are adjusted and fine-tuned during the training and testing phase to maximize the performance. The model is then saved into the system to be used in the later stage for the classification task. This step is shown in Algorithm 1 on line 31.

- Classification - The classifier model is then used to predict the output label (title, abstract, keyword, author, and other) for the input node contents for the new PDF document. This step is shown in the Algorithm 2.

5.2.3 Subject Hierarchy Classification

Subject Hierarchy Classification falls under the category of document or general text classification. In simpler terms, document classification refers to the task of assigning a document to one or more classes or categories. As described in the Speech and Language Processing book [39] by authors Daniel Jurafsky and James H. Martin, the goal of the classification task is to take a single observation, extract useful features, and thereby classify the observation into one of the sets of

Algorithm 1 CRF Classifier Model Training

Require: ground truth dataset, raw PDF files

```
for each PDF do
    Perform Conversion PDF-to-XML
    Perform preprocessing XML-to-ModifiedXML
end for
for each xml file do
    for each textbox node do
        Tag label using groundTruth data
    end for
end for
Initialise list objects, Features and Labels
Initialise XML objects, currentNode and previousNode
for each xml file do
    for each textbox node do
        Assign textbox to currrentNode
        if textbox node contains data element then
            Pass (currrentNode , previousNode) to method GenerateFeatures
            Method - GenerateFeatures produce features object.
            Append Node features to Features
            Extract label and append to Labels
        end if
        Assign currrentNode to previousNode
    end for
end for
Split Features, Labels into Train/Test(80/20)
Initialise CRF model
Update model parameters like algorithm,c1, c2, max-Iterations, all-Possible-Transitions
Train CRF model on training data
Test CRF model on testing data
Model Performance Evaluation
Fine-tune model parameters to optimize Accuracy
Save the Model
```

Algorithm 2 Linear Chain CRF Classifier

Require: XML object, pre-trained model
Ensure: Metadata: XML object with metadata labels

```
Input XML object
Initialise XML objects, currentNode and previousNode
for each textbox node in XML object do
    Assign textbox to currentNode
    if textbox node contains data element then
        Pass (currentNode, previousNode) to method GenerateFeatures
        Method - GenerateFeatures produce features object.
        Pass object features to trained CRF Tagger
        Update CRF output label to currentnode
    end if
    Assign currentNode to previousNode
end for
Return XML object with metadata labels
```

discrete classes. A simpler classifier could be developed using rule-based heuristics [48], but rule-based classifiers do not generalize well in most cases due to static rules, changes in data over time, and unseen new data. Another approach to building a classifier is to use supervised learning where input observations are associated with correct labels. Formally, the task of supervised classification can be defined to take an input x and a fixed set of output classes $Y = y_1, y_2, ..., y_m$ and return a predicted class $y \in Y$.

The subject hierarchy classification task uses a multinomial Naive Bayes (NB) classifier model [42,49,50] based on Bayes Theorem [51] available in scikit-learn [27] library for generating an implicit list of subject headings based on PDF documents. The Bayes theorem makes two assumptions: the feature's position has no significance and is conditionally independent of other features, allowing the classifier to make predictions quickly and accurately. The bag-of-words approach [52] is considered for the first assumption where the position is irrelevant, and the frequency of occurrence of each word is used as a feature for training a classifier. Following is the Naive Bayes equation :

$$P(C|X) = \frac{P(X|C) \cdot P(C)}{P(X)} \tag{3}$$

In this equation:

- X : Consists of input features $x_1, x_2, ..., x_i$

- C : Consists of fixed labels $c_1, c_2, ..., c_k$
- $P(C|X)$: Conditional probability of class C given the input feature X
- $P(X|C)$: likelihood for observing the feature X given the class C. e.g. $P(x_1, x_2, ..., x_i|c)$
- $P(C)$: The prior probability of class C. It represents the probability of a document belonging to class C based on historical data or prior knowledge
- $P(X)$: Termed as evidence, it is the probability of observing the features X across all possible classes. It serves as a normalizing factor to ensure that the sum of probabilities over all classes is equal to 1

Considering the feature independence assumption, the likelihood probability for observing features $x_1, x_2, ..., x_i$ given a class c is simplified as :

$$P(x_1, x_2, ..., x_i|c) = P(x_1|c) * P(x_2|c) * P(x_3|c) * ...P(x_i|c) \tag{4}$$

Text Normalization - Lemmatization and Tokenisation

Text Normalisation is the first step in the classification task. It accepts XML node contents as input in the document pre-processing step. In the text normalization steps, common english words known as stop words are removed along with special characters and punctuation. Lemmatization is performed to reduce words to their base form similar to previous ASHG 1 [9]. Sentence and Word tokenizers are used from NLTK [28] library for converting raw text data into sentence fragments and word tokens. The pre-processed tokens are then passed to the next step of the classification task.

Reduction of Controlled terms

This step is similar to previous ASHG 1 [9] where weights are assigned to the terms, such as high weight for deemed important terms and low weight for less important words. ASHG's controlled term favors the terms with low frequency in the ASHG's subject headings over those with high frequency. For example, terms such as algorithm, design, system, application, etc, are

common technical words with a high frequency of occurrences in the research paper's contents. Considering such words will impact the overall weighting scheme, potentially overshadowing rare yet highly relevant terms specific to the PDF document, thereby impacting the classifier model output. The ASHG 2 extractor uses a priority-based approach to tackle terms with high frequency. If high-frequency terms appear in the metadata elements such as keywords, abstract, and title, the module considers these terms for classification as they were added by the paper author, signifying high relevancy to their research, and vice versa if not present in the metadata section and present in the research paper body contents, it ignores these terms for generating the implicit list. The list of controlled terms with high-frequency occurrence and lower weights in the research/academic document is provided in the Appendix section.

ConfSys Subject Hierarchy Terms

This step is similar to the previous ASHG 1 [9], using ConfSys subject hierarchy terms derived from ACM's computing classification system [53]. These subject hierarchy terms are class labels for the model training and classification task. In ConfSys4, the subject hierarchy terms are revised to include more terms as per the latest classification provided by ACM [53].

Model Classification

The labeled dataset for training is taken from the past events data hosted on the ConfSys system and its tagged subject headings. In data preparation, text normalization is performed to convert raw document contents to lemma token forms. Using a bag-of-word approach, the frequency or occurrence count of the controlled terms associated with ConfSys4 subject hierarchy terms across the document contents is normalized and used as a feature for the model. The model is trained on the normalized term frequency counts and the associated output label from labeled data for training. The model performance is evaluated using the accuracy score as a performance metric for each predicted class label on the testing data. Once the model is trained and tested with the labeled data, the classification task for the new input PDF document returns the implicit list of subject hierarchy terms.

5.2.4 Author Profile Generation

Author Profile Generation is an important step in the metadata extractor system. It involves identifying and extracting information related to the authors of a paper, such as author name, email addresses, organization, location, and affiliations. It involves the use of the gazetteer data [54], similarity ratio score, document layout reading order [55], regular expression [33] and named entity recognition with BERT Transformer models [31, 32], NLTK [28] and Spacy [29] for generating author profile for each author extracted from the PDF paper. Challenges involved in its extraction include variation in author information format inside scholarly papers, multiple document layouts, and correctly linking authors to their corresponding details.

The algorithm outlined in 3 provides a systematic approach to address these challenges. It takes the output of the CRF classifier nodes tagged with the label "author" as input and gazetteer data containing information related to system user details such as name, email, organizations, departments, and addresses, shown in lines 1 to 3 in Algorithm 3. The XML nodes are traversed, passed through gazetteer data, compared, and matched using the similarity ratio score described in Section 3.3.3 with a threshold limit of 0.7, and if matched, are tagged with the appropriate entity labels. If not matched, passed to BERT-NER model [32] from the hugging face machine learning library [30, 36] for tagging entity labels such as PER (Person), ORG (Organisation), LOC (Location), and MISC (Miscellaneous), shown in lines 4 to 21 in Algorithm 3. The named entity tagging for the author nodes is also performed separately with NLTK [28] and Spacy [29] for evaluating the performance of entity recognition by all three libraries such as BERT, NLTK, and Spacy. Once all the nodes are tagged, reading order logic is used to segregate author details for each author detected to generate author profiles, as shown in lines 22 to 23 in Algorithm 3. These author profiles are nested dictionary objects containing the author's name as key and details such as email, organization, location, and affiliation as their value. The author details is a dictionary object containing key-value pairs for each detail. The author profile, once generated, is validated to ensure all details captured are consistent across all its authors, as shown in lines 24 to 25 in Algorithm 3.

Algorithm 3 Author and its Details Extraction

Require: XML object, Gazetteer data

Ensure: Metadata: Author (dictionary)

```
Load XML nodes tag to label "author" for Input XML node. Refer Algorithm 2
Initialize dictionary object, author
Load Gazetteer data for system user name, organization, department, and address
for each textbox node do
    if textbox node contains data element then
        Pass data to method - TagGazetteer
        Method TagGazetteer outputs (label,data)
        if (label,data) Exists then
            Append (label,data) to tagEntity
            Continue to next node
        end if
        if (label,data) Not Exists then
            Pass data to method - NamedEntityRecognition
            Method - NamedEntityRecognition outputs (label,data)
            if (label,data) Exists then
                Append (label,data) to tagEntity
                Continue to next node
            end if
        end if
    end if
end for
Pass tagEntity to method GenerateAuthorProfile
Method - GenerateAuthorProfile outputs object author
Pass author to method VerifyAuthorProfile
Method - VerifyAuthorProfile checks, verify, update and output object author
Return Metadata: Author (dictionary)
```

5.2.5 Author References Processing

This step involves processing nodes tagged as ”other” by the CRF classifier. It traverses XML and uses a rule-based search for node contents containing the phrase References. If matched, node attributes such as font type, font size, and bounding box are stored and used for further processing in extracting reference citations present in the later XML nodes. Based on the heuristic function, reference citations are extracted from the XML nodes. Each citation text extracted is then passed to the named entity detection method to identify PERSON-tagged text elements. The named entity detection method performs entity recognition using BERT NER model [32], NLTK [28], and Spacy [29]. All the tagged text nodes are verified to restrict duplicate authors and then consolidated to generate an implicit list of author names extracted from the reference section of the PDF document.

5.2.6 Validation Checks on Extracted Metadata

The following checks are implemented on extracted data to generate a reliable semantic header profile for an information resource by the ASHG 2 extractor :

- Data integrity checks to identify missing values in metadata objects and replace them with unknown or blank.
- Data normalization checks for author profiles to ensure author-related details such as email, organization, location, and affiliation are captured for each extracted author.
- Data quality checks to identify and remove noisy or dirty data or stop words such as white spaces and common english words.
- Identify and drop duplicate details in metadata objects in case of author metadata extraction

5.3 ConfSys4 : Contribution

- Implementation of modules such as document processing, information retrieval, and document classification in Python language.
- Integration of Python implemented modules with legacy Java Servlet code.

- Incorporating communication between Java Servlets and Python interfaces for the newly implemented modules.
- Java Servlets and JSPs are modified to handle new feature implementations such as metadata extraction for the submitted PDF document, detecting duplicate submission and invalid PDFs, comparing and matching author details with system user details for improving single/double/triple-blind, reminder email routine for program committee members for updating their topic of expertise for better paper allocation. Furthermore, additional feature implementation scripts (Java Servlet and JSPs), such as the author pairs identification with conflicting interests using DBLP data to leverage the similarity ratio score derived from Levenstein distance (edit distance), improvement to the PayPal payment interface, and the PDF document generation for the invoice, payment receipt for user registration, and final/preliminary program schedule are modified.
- Extraction support for metadata objects such as title, abstract, keyword, author details, and author references from the PDF format document.
- Includes supervised machine learning algorithms such as linear chain CRF classifier along with rule-based heuristics for classification and predicting metadata objects
- Includes probabilistic classifiers such as Naive Bayes for subject hierarchy classification of PDF document
- Uses additional features such as font type, font size, geometric position (bounding box), and relative position (X, Y) in addition to textual-based features for classifying metadata objects
- Includes NER extraction techniques using BERT models [31], NLTK [28] and Spacy [29] for author profile generation and author references processing

5.4 ConfSys4 : Limitations

- Extraction of tabular data in PDF document is not supported
- Extraction of image data in PDF document is not supported

- PDF documents with unknown or unsupported encodings are not supported. Text content within a PDF, with encodings such as ASCII, and UTF-8 is supported

Chapter 6

Experiments and Results

6.1 Overview

This chapter provides detailed information about the experimentation performed with ASHG 2 for testing and evaluating its performance for metadata extraction and document classification for ConfSys4 features such as metadata generation, single/double/triple-blind review, duplicate submission, paper submissions, and PDF processing.

6.2 Experimentation and Results

In dealing with natural language problems, the performance evaluation of the language models uses extrinsic and intrinsic evaluation as described in the Speech and Language Processing book [39]. Extrinsic evaluation involves testing the application's performance where the model is deployed. It helps in understanding whether the model implementation into the system has benefited the application, for example, language translation applications. Extrinsic evaluation requires language model development and testing on the application with no certainty of model performance for response time or output result. Intrinsic evaluation tests the model performance independent of the application. For example, the output of the classification task is compared and tested against baseline labeled data. Both extrinsic and intrinsic methods are incorporated to evaluate the performance of ConfSys4's metadata extraction system ASHG 2.

6.2.1 Extrinsic Evaluation

In extrinsic evaluation, the ASHG 2 is incorporated into the ConfSys system. The metadata generation across multiple PDF document layouts is tested in order to verify its integration into the ConfSys4 system and its performance for metadata extraction and classification. The following tests were performed in extrinsic evaluation to verify the robustness of the ConfSys4 system's metadata generation :

(1) Duplicate paper submission based on extracted metadata objects to restrict multiple submissions

(2) Empty PDF submission based on document size to restrict irrelevant PDF uploads during submissions

(3) Invalid file format check to enforce submission to only PDF formatted documents

(4) PDF document integrity check to restrict support for metadata details in images, tabular, and unsupported encoding format

(5) Submitting user details verification in the PDF author metadata section to ensure the submitting user should be one of the paper authors. Only the user with Admin and General Chair role is allowed to submit papers for other authors

(6) Author details verification with system users to identify new authors and request for preliminary sign-up before submission

(7) Verification of paper submission for single-blind review to ensure paper authors are not members of any organizing committee

(8) Verification of paper submission for double-blind review to ensure paper author details are not present in the author metadata section in the PDF document

(9) Verification of paper submission for double-blind review to ensure references for paper authors who are members of the organizing committee are not included in the PDF document

(10) Verification of paper submission for double-blind review to ensure review selection is updated to triple-blind if paper authors belong to Track Chair or Program Chair or General Chair role.

Testing User Interactions with ConfSys4 Interface

User interactions with the ConfSys4 interface for functionalities such as paper submission, blind review selection, paper allocation, conflicting author identification, payment processing, and PDF document generation are tested for multiple user input scenarios in extrinsic evaluation testing. ConfSys4's response to these interactions is recorded to understand its behavior, incorporate fail-safe mechanisms, and handle exceptions. Table 6.1 illustrates user interaction cases and appropriate system responses. The system rejects the PDF document with slight modifications in the paper-related metadata details submitted for the second time as a new submission entry. The similarity ratio score described in Section 3.3.3 is calculated for the uploaded PDF metadata details and compared with already submitted papers. If the similarity score is greater than 0.7, the system associates it as a duplicate entry and rejects the paper submission. The PDF document with major modifications in the paper-related metadata details submitted is accepted as a new submission entry only if the similarity ratio score is less than 0.7 for its metadata objects, such as title, abstract, and keywords. In cases where a user uploads a PDF document with no metadata details, unsupported encodings, or metadata details in tabular and image format, the system rejects the submission and informs the user to submit a valid PDF document for submission. The review selection by the user for the submitted PDF is considered to verify author metadata and reference sections present in the PDF. In case of double-blind selection, if a user uploads a PDF document with author metadata details, the system rejects the submission and informs the user to remove author details from the metadata section. The user with roles Admin/General Chair is only allowed to submit PDFs for other users. In case an organizing committee member uploads a PDF for other authors, the system rejects the submission. The double-blind submission is automatically converted to triple-blind if the user adds author details associated with Admin/General Chair/Program Chair roles.

Test Cases	User Interaction Examples	ConfSys4 Response Results
Duplicate Paper Submission	minor modification in metadata details such as paper title, abstract, and keyword	submission is rejected
Duplicate Paper Submission	major modification in metadata details such as paper title, abstract, and keyword	submission is accepted
Invalid PDF	upload PDF with no paper-related metadata details	submission is rejected
Invalid PDF	upload PDF with unsupported encoding format	submission is rejected
Invalid PDF	upload PDF with images data for metadata details	submission is rejected
Invalid PDF	upload PDF with tabular data for metadata details	submission is rejected
Empty PDF	upload [0-2] KB PDF file	submission is rejected
Invalid File Format	upload any other format file	submission is rejected
Paper Submission by Normal User	author metadata section with submitting user details	submission is accepted
Paper Submission by Normal User	author metadata section without submitting user details	submission is rejected
Paper Submission by Organizing Committee Members for other authors	paper submission by Admin/ General Chair for other authors	submission is accepted
Paper Submission by Organizing Committee Members for other authors	paper submission by program chair/track chair/program committee role for other authors	submission is rejected
Double Blind Submission	paper without author metadata section	submission is accepted
Double Blind Submission	paper with author metadata section	submission is rejected
Double Blind Submission	paper with reference citation to authors of the paper not associated with organizing committee	submission is accepted
Double Blind Submission	paper with reference citation to authors of the paper associated with organizing committee	submission is rejected
Triple Blind Submission	Double Blind Submission with one or more authors associated to Admin/General Chair/Program Chair	submission is accepted

Table 6.1: Testing User Interactions with ConfSys4 Interface

Table Name	Details
pdf_paper_details	list of PDFs for metadata generation
pdf_paper_metadata	metadata details such as title, abstract, keyword extracted from list of PDFs in pdf_paper_details
pdf_paper_author_details	author metadata details name, email, org, and location extracted from list of PDFs in pdf_paper_details
pdf_paper_subject_hierarchy	subject heading extracted and associated to the list of PDFs in pdf_paper_details

Table 6.2: Metadata extraction results recorded and stored in ConfSys1

6.2.2 Intrinsic Evaluation

In intrinsic evaluation, ground-truth metadata from previously managed events is considered for testing the ASHG 2 metadata generation results. The labeled data includes paper-related details such as title, abstract, keyword, subject hierarchy, author-related details, and raw PDF documents for processing and information extraction. Approximately 300 PDF documents were considered for the evaluation. The following steps were performed for intrinsic evaluation :

(1) Load labeled metadata for 300 PDF files

(2) Metadata extraction using the ASHG 2, and other extractors such as Cermine [12, 13] and Grobid [14].

(3) Benchmark the extractor system results against baseline labeled data using the similarity ratio score described in Section 3.3.3.

(4) Performance evaluation using accuracy to measure the quality of the extractor results.

(5) Performance evaluation using accuracy for named entity recognition for author details extraction using NLTK [28], Spacy [29] and BERT-NER [32] models [32] for ConfSys4.

Table 6.2 provides details for the intrinsic evaluation performed in ConfSys4. The list of PDF documents used for evaluation is present in table "pdf_paper_details". Paper-related metadata details such as title, abstract, and keyword are extracted and stored in the table "pdf_paper_metadata" for all the PDFs. Author-related details are recorded and stored in the tables "pdf_paper_author_details". Details such as name, email, organization, and location are extracted for each author present in the

```
yogesh.yadav@U-37D5LNSGU589I:/usr/local/tomcat/webapps/ConfSys1/python/code$ python
/usr/local/tomcat/webapps/ConfSys1/python/code/Metadata_Extractor_Load.py --fn "ConfSys4-ICMS" --pdf_dir
"/usr/local/tomcat/webapps/ConfSys1/python/pdf" --xml_dir "/usr/local/tomcat/webapps/ConfSys1/python/xml"
--mod_xml_dir "/usr/local/tomcat/webapps/ConfSys1/python/output"
-----------------------------------------------------------------
Title is CONFSYS - AN INTELLIGENT CONFERENCE MANAGEMENT SYSTEM
-----------------------------------------------------------------
Abstract is THIS PAPER OFFERS A BRIEF HISTORY OF, CONFSYS, A CONFERENCE MAN- AGEMENT SYSTEM, THAT HAS BEEN USED FOR
OVER 15 YEARS TO SUPPORT A NUMBER OF INTERNATIONAL ACADEMIC CONFERENCES. IT IS A COMPLETE SYSTEM THAT HAS ALL
FUNCTIONS AUTOMATED WITH THE POSSIBILITY OF THE PROGRAM CHAIR OVERRIDING ANY OF ITS DECISION. WE HAVE FOUND THAT IN
MOST INSTANCES, THE DECISIONS MADE BY THE SYSTEM NEED VERY MINOR CHANGES. THIS PAPER DESCRIBES ANOTHER STEP IN ITS
AUTOMATION PRO- CESS INVOLVING THE SUBMISSION MADE BY AUTHORS AND ITS PROCESSING BY A PROPOSED INTELLIGENT MODULE.
THE NEW MODULE WILL EXTRACT THE SALIENT METADATA WHICH WE BELIEVE ARE MORE RELEVANT THAN THE ONES ENTERED BY
AUTHORS. THIS WOULD ENSURE RELIABLE PAPER-RELATED DETAILS LIKE TITLE, AUTHOR, COAUTHOR, ORGANIZATION, , KEYWORD,
ETC. ARE BEING CAPTURED INSTEAD OF USERS ADDING THESE DETAILS FIRST-HAND. THE SYSTEM REQUIRES USERS TO VERIFY THE
EXTRACTED INFORMATION AND CORRECT THEM IF REQUIRED, FURTHER IMPROVING THE PAPER ALLOCATION PROCESS TO REVIEWERS
BASED ON MATCHING THE REVIEWERS INTERESTS WITH EXTRACTED  AND , THUS IMPROVING THE QUALITY OF RELEVANCE OF THE
REVIEWS AND COMMENTS TO THE AUTHORS. THIS IN TURN WOULD IMPROVE THE QUALITY OF THE PUBLICATIONS.
-----------------------------------------------------------------
Keywords is CONFERENCE MANAGEMENT SYSTEM(CMS), INTELLIGENT SYSTEMS, PA- PER SUBMISSION, PAPER ALLOCATION, RULE-BASED
HEURISTICS, SUPERVISED CLASSIFICATION, AND INFORMATION COMMUNICATION TECHNOLOGY (ICT).
-----------------------------------------------------------------
Author-Affiliation is {"Yogesh O. Yadav": {"ORG": ["Concordia University"], "GPE": ["Quebec", "Montreal", "Canada"],
"EMAIL": ["yogeshoyadav08@gmail.com"], "AFFILIATION": ["unk"]}, "Bipin C. Desai": {"ORG": ["Concordia University"],
"GPE": ["Quebec", "Montreal", "Canada"], "EMAIL": ["bipinc.desai@concordia.ca"], "AFFILIATION": ["unk"]}}
```

Figure 6.1: Metadata Extraction Result from ASHG 2 using command line utility

PDF document. Subject headings extracted based on the contents of the PDFS are recorded and stored in the table "pdf_paper_subject_hierarchy".

Metadata Extraction with ASHG 2, Cermine, and Grobid

The metadata extraction using ASHG 2 offers a command line utility to extract metadata elements such as title, abstract, keyword, subject hierarchy classification, and author metadata, including name, email, organization, and location for the input PDF file. It is integrated into the ConfSys4 system and offers web service calls to generate and display extracted metadata for the input PDF submitted during the paper submission process. The extraction result using the command line utility of ASHG 2 is shown in Fig 6.1.

Cermine [12, 13] and Grobid [14] are publicly available extractor tools that offer batch processing and web service calls for input PDFs for metadata extraction. These systems generate metadata elements such as title, abstract, keyword, and author metadata from the input PDF. The ASHG 2 classifies the input PDF into subject classifications based on the contents of the PDF. The classification into categories is not available with these public extractor systems. Figs 6.2 and 6.3 show the metadata extraction result for the input PDF using the web service utility available for the Cermine and Grobid systems. Web service utility in Cermine requires PDF document upload over the URL

Figure 6.2: Metadata Extraction Result from Cermine using web service call

(http://cermine.ceon.pl/index.html). It processes the PDF document, extracts metadata, and displays the metadata results, as illustrated in Fig 6.2. In Grobid, web service calls (https://kermitt2-grobid.hf.space/) require PDF document uploads and outputs metadata results in XML formatted files, as illustrated in Fig 6.3.

6.2.3 Performance Evaluation using Accuracy

The similarity ratio score described in Section 3.3.3 is used to compare extractor results for metadata objects such as paper titles, abstracts, keywords, and author details against the baseline ground-truth metadata available. The similarity score [0.6 - 0.7] is considered as threshold score for benchmarking. If the score is greater than 0.7, it is assigned a score of 1, and vice versa, it is assigned a score of 0. Accuracy is calculated as the ratio of correctly predicted instances to the total instances evaluated. The following equation is used to calculate the accuracy for each metadata

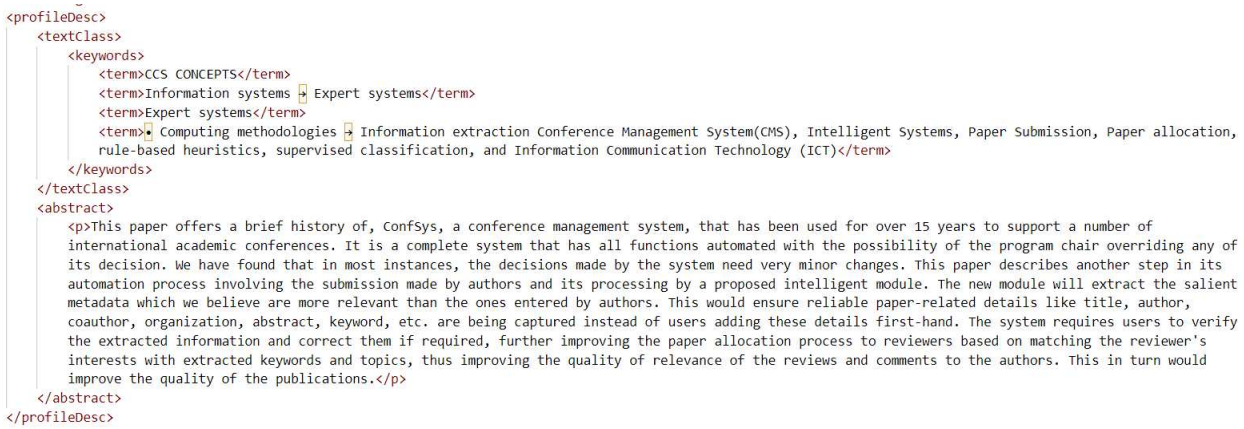

Figure 6.3: Metadata Extraction Result from Grobid using web service call

object, as the ratio of correct predicted instances to total metadata instances.

$$Accuracy = \frac{NumberofCorrectPredictions}{TotalNumberofPredictions} \tag{5}$$

Fig 6.4 illustrates accuracy results for metadata objects such as title, abstract, author name, and keywords for ASHG 2, Cermine [12, 13], and Grobid [14]. The ASHG 2 has an overall accuracy of 0.80 for all metadata objects and performs better in the extraction of keyword and author detection than its counterparts. The abstract detection results show better performance by Cermine (0.93) in comparison to ASHG 2 (0.90). Both systems, in consideration, have used different models and feature selection during the model training phase. ASHG2 model training and feature selection can be improved for abstract detection as a future work. The accuracy for title extraction is 1 across all extractor systems, signifying that the metadata title has a fixed format in the document structure across the document layouts. For example, the paper title has maximum font size and is present on the first page at the top of the document.

6.2.4 Named Entity Recognition (NER)

Furthermore, in intrinsic evaluation, the performance of named entity recognition for author metadata elements such as name, organization, and location using libraries such as Natural Language Tool Kit (NLTK) [28], Spacy [29], and BERT-NER models [32] is evaluated against baseline

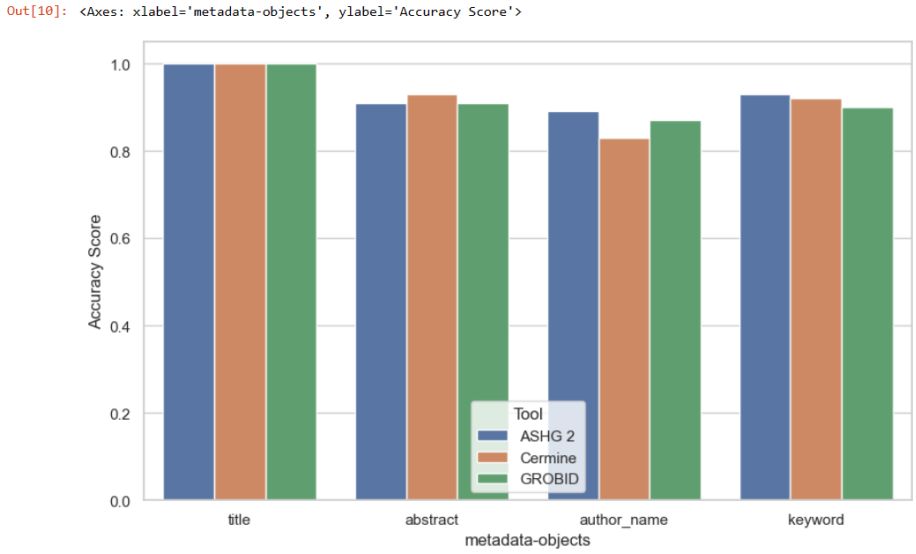

Figure 6.4: Accuracy score for Metadata Extraction

labeled dataset. The NER task implementation with the NLTK library did not produce satisfactory results for author name detection. To improve the performance, the NER task with the Spacy library was implemented. The performance improved in comparison to NLTK, but it did not generalize enough to accurately detect the variety of author names present in the PDFs involved in testing. This led to using a transformer model from hugging face [30], a pre-trained BERT-NER models [31,32] designed for tagging Person, Organisation, Location, and Miscellaneous to the wide variety of input text from the unstructured PDF. The BERT-NER model significantly improved the accuracy of author metadata extraction for author profile generation. Author metadata elements such as name, organization, and location for approximately 300 PDFs are generated using NLTK, Spacy, and Bert-NER, respectively, and stored in the system to compare and evaluate performance against labeled ground-truth data.

Fig 6.5 illustrates accuracy results for named entity objects using the NLTK [28], Spacy [29], and Bert NER Models [32]. The overall accuracy of above 0.90 is achieved by the BERT-NER across metadata elements such as author name, organization, and location as compared to NLTK and Spacy. The author names detection shows significant improvement with BERT-NER having an accuracy score of 0.9 in comparison to Spacy (0.78) and NLTK (0.7). Thus, BERT-NER models generalize well and are correctly able to detect author names for a variety of PDFs involved in the

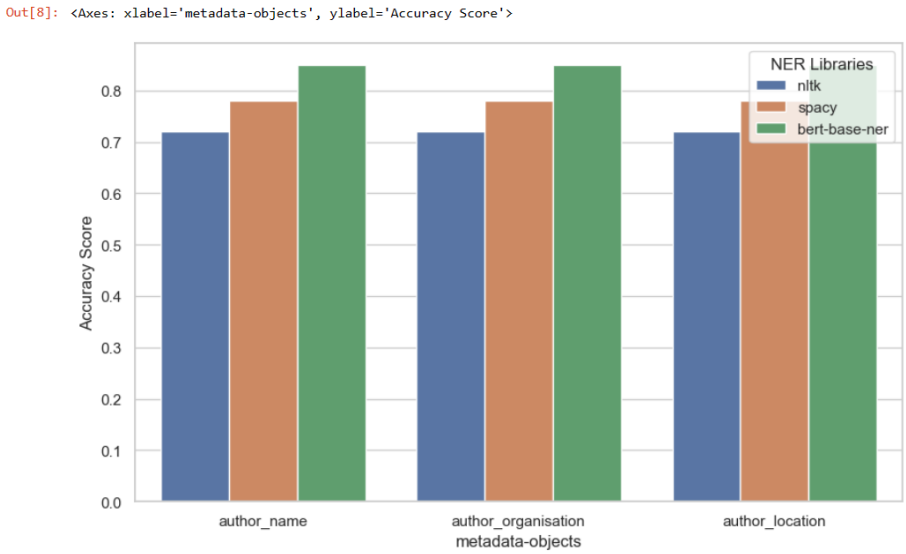

Figure 6.5: Accuracy score for author name, organization, and location extraction testing.

Chapter 7

Conclusion and Future Work

7.1 Contribution to ConfSys System

In this thesis, we presented the ConfSys4 system and implementation of the metadata extraction system (ASHG 2) for document processing, metadata generation, and document classification. The ASHG 2 integration into the ConfSys4 system has improved the functionality of the paper submission, single/double/triple-blind, and paper allocation process in managing conferences, thus improving the overall experiences of users and organizing committee members and, in turn, the quality of the publication/event. The ConfSys4 platform, in addition, has improved existing functionality for features such as adding automatic reminder emails to organizing committee members, improved author-pair identification for conflicting users, improved Paypal payment interface for user registration to events, and PDF generation for invoice, payment receipt and generated program for the event.

The ASHG 2 extractor system presented in this thesis is implemented by using rule-based heuristics and classification algorithms such as Conditional Random Fields (CRF) and Naive Bayes (NV), and natural language processing techniques for document processing, information retrieval, and document classification for reliable Semantic Header Generation. The extractor system is developed completely in Python language and is integrated with Java Servlet/JSPs in the ConfSys4 system. The Python libraries used for the implementation of ASHG2 include Sciket-learn [27],

Hugging Face Transformers [30, 36], PDFMiner [35], LXML [34], Levenstein [56], Natural Language Tool Kit (NLTK) [28], Spacy [29] and BERT-NER Models [31, 32].

The extrinsic evaluation and testing performed on the ConfSys4 platform ensures the reliable generation of semantic header profiles for the submitted document and only involves user input for metadata validation and verification, thus making the ConfSys4 platform competitive across other CMS systems [23–26] for managing academic events.

7.2 FutureWork

7.2.1 ORCID Unique Identifier Integration into ConfSys

ORCID(Open Researcher and Contributor ID) [57] is a digital identifier for researchers and scholars in the research/academic community. It serves as a unique and unambiguous identifier, connecting individuals with their scholarly activities and outputs. This ORCID identifier remains constant throughout an individual's academic and professional career, regardless of name changes, affiliations, or research areas.

The ConfSys system [1–4, 58] uses email addresses as a unique identifier for signing up the user into the ConfSys system. It further requests users to add details such as personal details, organizations, departments, addresses, and topics of interest to collect more information about the user. In the ConfSys4 system, users can have multiple accounts by signing up with different email addresses. This leads to duplicate records for the same user in the system. In order to improve the ConfSys4 system, ORCHID ID can be considered to uniquely identify each user for creating the account. This will restrict users from having multiple accounts throughout the lifespan of their academic or professional careers.

7.2.2 Extraction support to Tabular contents and Image data

The research articles often see tabular data showing comparison statistics or images showing workflow or architecture diagrams. Extraction of tabular data or images is currently not supported in the ConfSys4 system. The ASHG 2 in the ConfSys4 system is focused on semantic header profile generation using a primary set of metadata elements essential for discovering/searching an

information resource. The tabular-based information or image contents present in the underlying PDF document act as secondary information about an information resource. If extracted from the PDF documents, the tabular or image contents can capture additional layers of detail and visual representations such as comparison statistics, workflow diagrams, architectural representations, or other visual and structured data essential for the research paper. In the ConfSys4 systems, Program Committee members reviewing the paper submitted during the event can be provided with this secondary information. This will enable them with a richer understanding of the research content for reviewing that is beyond the primary textual metadata.

7.2.3 Originality Score by Independent Organization

The publishers involved in publishing the conference proceedings have requested originality scores (plagiarism detection score) for the accepted papers in recent years. The originality score of a paper is determined by analyzing the content of the paper for similarities with existing works to assess its uniqueness and authenticity. Several third-party systems offer plagiarism detection services that provide originality scores and reports by reviewing the content of the paper and comparing it against scholarly database contents, including open-access journals, books, conference proceedings, and other academic content items. API calls to these independent systems can be integrated into the ConfSys4 system for the submitted papers and record the output response from these systems containing the originality score and report into the system, thus enabling the evaluation of submitted papers for originality. This integration will further streamline the assessment process, ensuring that originality scores and detailed reports are efficiently captured and managed alongside the submitted papers within the ConfSys4 platform.

Appendix A

A.1 Controlled Terms Exclusion

As described in section 5.2.3, Table A.1 shows the list of controlled terms that are assigned lower weight. These terms occur frequently in the research/academic paper and impact the classifier model to classify the paper to these controlled terms. To prioritize rare words that occur less frequently but have high weightage for the paper classification, it is essential to not consider these high-occurring controlled terms for subject classification for the PDF document classification task.

A.2 Extrinsic Evaluation Testing

Table 6.1 shows user interactions with the ConfSys4 system for extrinsic evaluation. The duplicate submission check ensures slight modifications of the metadata details such as title, abstract, or

Controlled Terms		
algorithm	experiment	model
analysis	file	network
application	function	organization
architecture	general	other
constraint	information	performance
copyright	learning	physics
data	logic	power
design	manage	reference
engineering	measure	server
evaluation	metric	software
miscellaneous	system	

Table A.1: Controlled Terms - High Frequency Occurrence and Low Weightage

```
Operation Post Parsing : paper_metadata
Submitting Author is Yogesh O. Yadav
Similarity for Extracted Title: 0.803030303030303
Similarity for Extracted Abstract: 0.7366771159874608
```

Figure A.1: Similarity Score for Modified PDF submission

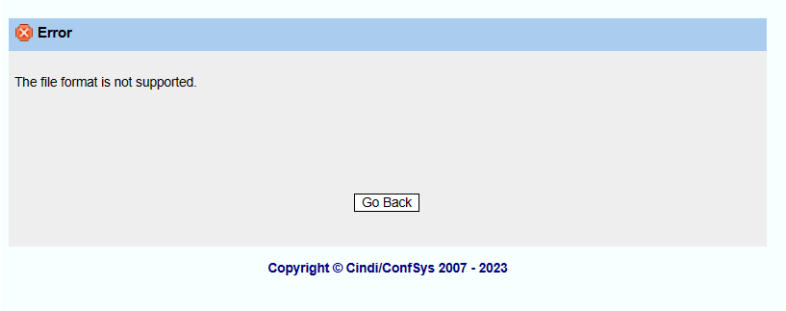

Figure A.2: Invalid Document Format Submission Error

keyword in the PDF, if uploaded again by the user, will reject the paper for submission. In order to test slight modifications in paper metadata, two PDF documents are used. The first PDF document has the title ”ConfSys - An Intelligent Conference Management System,” and the second PDF has the updated title ”ConfSys - An Intelligent Conference Management System-BCD-DB-NA.WR”. If the second PDF is submitted after the first PDF, it will cause a duplicate submission check rejection due to the similarity ratio score. The similarity ratio score described in Section 3.3.3 for the second PDF is greater than 0.7, as shown in Fig A.1 when compared and matched with the first PDF. The user submitting the paper if uploads a document with a format other than PDF, empty PDF, or PDF with an unsupported encoding format, the ConfSys4 system will reject the PDF with the message shown in Fig A.2.

```
/usr/local/apache-tomcat-9.0.37/webapps/ConfSys1/python/model
[y_yadav@confsys1 model]$ ls -l
total 0
drwxrwsrwx. 1 y_yadav tomcat 56 Aug 20 00:30 bert-base-NER
drwxrwsrwx. 1 y_yadav tomcat 56 Aug 20 00:30 bert-large-cased-finetuned-conll03-english
drwxrwsrwx. 1 y_yadav tomcat 56 Aug 20 00:30 custom-classifier
drwxrwsrwx. 1 y_yadav tomcat 56 Aug 20 00:30 distilbert-base-uncased
```

Figure A.3: Saved Model Details in ConfSys System

Scripts	Details
java/confsys3/daemon/DBLPUpdater.java	Code revised to dynamically process DBLP data using similarity ratio score described in Section 3.3.3
java/confsys3/daemon/Reminder.java	Code revised to share automatic reminder emails to organizing committee member for topic of interests
java/confsys3/menu/ContentSet.java	Code revised to handle data for ConfSys4 database initiation for new conference
java/confsys3/menu/MenuItem.java	Code revised to handle daemon refresh for admin role
java/confsys3/web/UploadParser.java	Code revised to incorporate ASHG2 metadata extraction
java/confsys3/web/servlet/conference /ConferenceRegisterControl.java	Code revised to add method for download PDF for invoice and payment recipets
java/confsys3/web/servlet/conference /PaperControl.java	Code revised to incorporate ASHG2 metadata generation
java/confsys3/web/servlet/conference /SessionControl.java	Code revised to add method for download PDF for generated program
jsp/conference/paper/paper_add.jsp	Code revised to add mode selection capability for author
jsp/conference/paper/paper_edit.jsp	Code revised to add scroll capability to subject headings selection for author
jsp/conference/paper/paper_subject_list.jsp	Code revised to add show subject headings with scroll capability
jsp/conference/registration_history_detail.jsp	Code revised to add PDF download button
jsp/conference/registration_payment.jsp	Code revised to incorporate Paypal Payment changes
jsp/conference/session_generate_program.jsp	Code revised to add PDF download button
jsp/header.jsp	Code revised to incorporate header changes in ConfSys4
jsp/bottom.jsp	Code revised to dynamically show current year

Table A.2: Modified Java Servlet/JSPs scripts in ConfSys4

```
[y_yadav@confsys1 code]$ pwd
/usr/local/apache-tomcat-9.0.37/webapps/ConfSys1/python/code
[y_yadav@confsys1 code]$ ls -l
total 196
-rwxrwsrwx. 1 y_yadav tomcat 75251 Jan 11 14:12 extractor.py
-rwxrwsrwx. 1 y_yadav tomcat 74999 Jan 11 13:08 extractor_v1.py
-rwxrwsrwx. 1 y_yadav tomcat  4153 Dec 18 19:12 Metadata_Extractor_Load.py
-rwxrwsrwx. 1 y_yadav tomcat  3858 Dec 18 19:12 Metadata_Extractor.py
-rwxrwsrwx. 1 y_yadav tomcat 18313 Dec 18 19:12 named_entity_detection.py
-rwxrwsrwx. 1 y_yadav tomcat   292 Dec 18 19:12 pdf_to_xml.py
-rwxrwsrwx. 1 y_yadav tomcat  4715 Dec 18 19:12 preprocess_xml_data.py
drwxrwsrwx. 1 y_yadav tomcat   618 Jan 11 14:14 __pycache__
[y_yadav@confsys1 code]$
```

Figure A.4: Metadata Extraction Script Details in ConfSys System

A.3 ConfSys4 - ASHG2 System Implementation Details

As described in sections 5.2.2 and 5.2.3, the following Fig A.3 shows saved model details after training and testing for classifier models such as Conditional Random Field (CRF) [41] and Naive Bayes (NB) [42], entity detection models such as Bert-NER [32] implemented in ASHG module for classification and prediction task. Fig A.4 shows Python scripts implemented in the ConfSys4 system to extract metadata elements and subject headings classification from the input PDF file.

A.3.1 Modified Java Servlet/JSPs scripts in ConfSys4

ConfSys4 Java Servlet and Jsp codes are modified to incorporate metadata extraction, subject heading classification, DBLP data processing, reminder emails for organizing committee members, single/double/triple blind selection, PayPal payment changes, and PDF document download features with respect to ConfSys3.5. In addition, communication between the Java servlet and Python interface for exchanging responses is incorporated in ConfSys4. The modified script and its details are illustrated in Table A.2.

A.3.2 New Java Servlet/JSPs scripts in ConfSys4

ConfSys4 incorporates the implementation of new scripts in the Java framework to support features implemented in the ConfSys4 system. The new code scripts added into ConfSys4 is shown in Table A.3

Scripts	Details
java/confsys3/daemon/DBLPCoAuthor.java	Includes code for identifying author pairs
java/confsys3/daemon/DBLPDownloader.java	Includes code for downloading DBLP file
java/confsys3/web/servlet/DaemonRefresh.java	Includes code for refreshing DBLP data
java/confsys3/web/UploadParser_Manual.java	Includes code for Manual Mode of Submission
java/confsys3/web/servlet/conference /PaperControl_Manual.java	Includes code for Manual Mode of Submission using ASHG2
jsp/conference/paper/paper_metadata_review.jsp	Includes code for displaying metadata contents for author verification
jsp/conference/paper /paper_metadata_review_manual.jsp	Includes logic for displaying metadata contents for author verification for manual mode of submission
jsp/conference/paper/paper_subject_list_manual.jsp	Include code for displaying subject headingsfor manual mode of submission
jsp/help/user/new_user_sign_up.jsp	Include code for new user signup
jsp/user/new_user_register_emails_detected.jsp	Include code for new user email detected
jsp/user/new_user_register_emails_detected_confirm.jsp	Include code for new user email confirmation

Table A.3: New Java Servlet/JSPs scripts in ConfSys4

Script	Details
python/code/Metadata_Extractor.py	Main execution script for calling methods such as pdf_to_xml.py, preprocess_xml_data.py, and extractor.py. Accepts PDF document as inputs
python/code/pdf_to_xml.py	Uses PDFMiner library for conversion of PDF to raw XML formatted documents
python/code/preprocess_xml_data.py	Uses LXML library for XML processing. Includes pseudo code for processing raw XML to pre-processed XML document described in Section 3.2
python/code/extractor.py	Uses Scikit-learn library for importing classifier models such as Condition Random Fields (CRF) and Naive Bayes(NB). Includes pseudo-code for algorithms 2 and 3 described in Section 5.2. Accepts pre-processed XML as inputs and returns extracted metadata objects
python/code/named_entity_detection.py	Uses NER libraries such as Natural Language Tool Kit (NLTK), Spacy, Hugging Face, Transformers (BERT), Levenshtein. Includes pseudo-code for detection of named entities such as name, email, organization, and location. Accepts text as input from extractor.py and returns named entity labels associated with text tokens.
python/model/	Stored model details for metadata extraction using ASHG 2
python/tokenizer/	Stored tokenizer details used for named entity recognition for BERT-NER models

Table A.4: Python scripts in ConfSys4

A.3.3 Python scripts in ConfSys4

ASHG 2 for metadata generation and subject hierarchy classification is implemented completely in Python using libraries such as PDFMiner [35], LXML [34], Scikit-learn [27], Natural Language Tool Kit (NLTK) [28], Spacy [29], Transformers [31], Hugging Face [30] and BERT-NER models [32]. The Python code implementation details are shown in Table A.4

A.3.4 Database Changes

ConfSys4 system incorporated changes into the ConfSys database with respect to ConfSys3.5. The subject headings records were added to the existing table as per the latest Classification Computing System (CCS) mapping present in the ACM [53]. The database changes details are shown in

Table Name	Details
content	New records added to allow admin to refresh daemons
dblp_archive	New table added to archive dblp data
dblp_log	New table added to log dblp data processing
menu	New records added to allow admin to refresh daemons
sendmail	New record added to send automatic reminder email to organizing committe members to update topic of interests
subject_hierarchy	New records added for subject headings by using latest Classification Computing System shared by ACM
sys_template	New records added for dynamic processing of DBLP dataset

Table A.5: ConfSys4 Database Changes

Table A.5

A.4 Python Libraries

It is recommended to use the Python 3.10 version. Following Python libraries need to be installed to setup the ASHG2 module into the ConfSys system

- nltk [28]
- spacy [29]
- pdfminer [35]
- lxml [34]
- scikit-learn, python-crfsuite and sklearn-crfsuite [27]
- pyTorch [37]
- transformers from hugging face [30, 36]
- levenshtein
- pandas
- MariaDB Connector/C

```
[y_yadav@confsys1 ConfSys_ICMS]$ pwd
/home/y_yadav/thesis/ConfSys_ICMS
[y_yadav@confsys1 ConfSys_ICMS]$ ls -l
total 488
-rwxrwxrwx. 1 y_yadav y_yadav  11973 Aug 19 23:07  ConfSys_Installation-README_Guide.md
-rwxrwxrwx. 1 y_yadav y_yadav  59652 Aug 19 23:07  ConfSys_Installation-README_Guide.pdf
-rwxrwxrwx. 1 y_yadav y_yadav  57094 Jun  5  2023 'ConfSysm ER Diagram Final.mwb'
-rwxrwxrwx. 1 y_yadav y_yadav  38672 Jun  5  2023 'Data Dictionary ConfSys System.xlsx'
-rwxrwxrwx. 1 y_yadav y_yadav  56547 Jun  5  2023 'Data Model ConfSys System.pdf'
-rwxrwxrwx. 1 y_yadav y_yadav 246797 Jun  5  2023 'Data Model ConfSys System.png'
drwxrwsrwx. 1 y_yadav y_yadav    352 Aug 19 23:07  documentation
drwxrwsrwx. 1 y_yadav y_yadav    130 Jun  5  2023  image
drwxrwsrwx. 1 y_yadav y_yadav   1634 Jun  5  2023  images
-rwxrwxrwx. 1 y_yadav y_yadav    139 Jun  5  2023  index.html
drwxrwsrwx. 1 y_yadav y_yadav    144 Aug 28 21:43  java
-rwxrwxrwx. 1 y_yadav y_yadav   7347 Nov  8 16:29  javafiles
drwxrwsrwx. 1 y_yadav y_yadav     56 Jun 30  2023  jdbc
drwxrwsrwx. 1 y_yadav y_yadav    278 Jan 11 21:54  jsp
drwxrwsrwx. 1 y_yadav y_yadav     44 Jan 11 22:10  letter-head
drwxrwsrwx. 1 y_yadav y_yadav     22 Jun  5  2023  META-INF
drwxrwsrwx. 1 y_yadav y_yadav     64 Aug 28 21:43  python
-rwxrwxrwx. 1 y_yadav y_yadav   1438 Jun  5  2023  README.md
drwxrwsrwx. 1 y_yadav y_yadav    360 Jun  5  2023  scripts
drwxrwsrwx. 1 y_yadav y_yadav    270 Nov  8 16:29  sql
drwxrwsrwx. 1 y_yadav y_yadav     38 Jun  5  2023  styles
drwxrwsrwx. 1 y_yadav y_yadav    246 Jun  5  2023  test
-rwxrwxrwx. 1 y_yadav y_yadav   1623 Jun  5  2023  _tomact_update.jsp
drwxrwsrwx. 1 y_yadav y_yadav    198 Sep  6 13:58  WEB-INF
[y_yadav@confsys1 ConfSys_ICMS]$
```

Figure A.5: Git Repository for ConfSys4

A.5 Git Release

ConfSys4 system is now available in Git on the ENCS server. The complete development of the ConfSys4 system, including the codebase for Java, JSP, and Python, commit history, releases, and associated documentation to facilitate version control, is implemented throughout the lifecycle in Git. The Git Repository path on ConfSys1 ENCS is shown in Fig A.5. A total of 85 commits were added to implement the ConfSys4 system.

Bibliography

[1] M. Huang, Y. Feng, and B. C. Desai, "Confsys: A web-based academic conference management system," in *Proceedings of the 2008 C3S2E Conference*, ser. C3S2E '08. New York, NY, USA: Association for Computing Machinery, 2008, p. 141–143. [Online]. Available: https://doi.org/10.1145/1370256.1370280

[2] ——, "Confsys2: An improved web-based multi-conference management system," in *Proceedings of the 2nd Canadian Conference on Computer Science and Software Engineering*, ser. C3S2E '09. New York, NY, USA: Association for Computing Machinery, 2009, p. 155–159. [Online]. Available: https://doi.org/10.1145/1557626.1557651

[3] M. Lu, K. Zhao, and B. C. Desai, "Confsys: A kaizen conference management system," in *Proceedings of the International Conference on Computer Science and Software Engineering*, ser. C3S2E '13. New York, NY, USA: Association for Computing Machinery, 2013, p. 5–13. [Online]. Available: https://doi.org/10.1145/2494444.2494488

[4] Z. Gu, X. Jin, and B. C. Desai, "Confsys: The cindi conference support system," in *Seventh International Database Engineering and Applications Symposium, 2003. Proceedings.* IEEE, 2003, pp. 414–418.

[5] "DBLP Dataset," https://dblp.org/.

[6] R. R. Prasath and P. Öztürk, "An approach to content extraction from scientific articles using case-based reasoning." *Res. Comput. Sci.*, vol. 117, pp. 85–96, 2016.

[7] E. Tonkin and H. L. Muller, "Keyword and metadata extraction from pre-prints." in *ElPub*, 2008, pp. 30–44.

[8] P. Flynn, L. Zhou, K. Maly, S. Zeil, and M. Zubair, "Automated template-based metadata extraction architecture," in *Asian Digital Libraries. Looking Back 10 Years and Forging New Frontiers: 10th International Conference on Asian Digital Libraries, ICADL 2007, Hanoi, Vietnam, December 10-13, 2007. Proceedings 10*. Springer, 2007, pp. 327–336.

[9] B. C. Desai, S. M. Haddad, and T. Wang, "Extracting semantics of documents using semantic header generator," *Concordia Spectrum Library*, 2008.

[10] J. Azimjonov and J. Alikhanov, "Rule based metadata extraction framework from academic articles," *arXiv preprint arXiv:1807.09009*, 2018.

[11] Z. Guo and H. Jin, "A rule-based framework of metadata extraction from scientific papers," in *2011 10th International Symposium on Distributed Computing and Applications to Business, Engineering and Science*, Oct 2011, pp. 400–404.

[12] D. Tkaczyk, "New methods for metadata extraction from scientific literature," *arXiv preprint arXiv:1710.10201*, 2017.

[13] D. Tkaczyk, P. Szostek, M. Fedoryszak, P. J. Dendek, and Ł. Bolikowski, "Cermine: automatic extraction of structured metadata from scientific literature," *International Journal on Document Analysis and Recognition (IJDAR)*, vol. 18, pp. 317–335, 2015.

[14] L. Romary and P. Lopez, "Grobid-information extraction from scientific publications," *ERCIM News*, vol. 100, 2015.

[15] H. Han, C. L. Giles, E. Manavoglu, H. Zha, Z. Zhang, and E. A. Fox, "Automatic document metadata extraction using support vector machines," in *2003 Joint Conference on Digital Libraries, 2003. Proceedings.* IEEE, 2003, pp. 37–48.

[16] K. Seymore and R. Rosenfeld, "Learning hidden markov model structure for information extraction," 1999.

[17] R. Chen, A. Perez, K. Dutta, and B. C. Desai, "Cindi: a digital library for academics," in *Proceedings of the 2nd Canadian Conference on Computer Science and Software Engineering*, 2009, pp. 1–6.

[18] B. C. Desai, "Supporting discovery in virtual libraries," *Journal of the American Society for Information Science*, vol. 48, no. 3, pp. 190–204, 1997.

[19] B. C. Desai, R. Shinghal, N. Shyan, and Y. Zhou, "Cindi: A system for cataloguing, searching, and annotating electronic documents in digital libraries," in *Foundations of Intelligent Systems: 11th International Symposium, ISMIS'99 Warsaw, Poland, June 8–11, 1999 Proceedings 11*. Springer, 1999, pp. 154–162.

[20] B. C. Desai, R. Shinghal, N. R. Shayan, and Y. Zhou, "Cindi: a virtual library indexing and discovery system," *Library Trends*, vol. 48, no. 1, pp. 209–209, 1999.

[21] C. Bipin, S. Sami, and A. Ali, "Automatic semantic header generator," in *Foundations of Intelligent Systems: 12th International Symposium*. Springer, 2010.

[22] S. Haddad and B. C. Desai, "Ashg: Automatic semantic header generator," *Master's thesis, Department of Computer Science, Concordia University, Montreal, Canada I*, vol. 998, 2000.

[23] A. Voronkov, "Keynote talk: Easychair," in *Proceedings of the 29th ACM/IEEE International Conference on Automated Software Engineering*, ser. ASE '14. New York, NY, USA: Association for Computing Machinery, 2014, p. 3–4. [Online]. Available: https://doi.org/10.1145/2642937.2643085

[24] "ConfTool Provided by ConfTool GmbH in Hamburg, Germany," https://www.conftool.net/en/index.html.

[25] I. M. Yassin, Y. M. Yusof, A. Johari, A. Zabidi, and H. A. Hassan, "Crs: Registration extension of the openconf™ conference management system," in *2011 IEEE International Conference on Computer Applications and Industrial Electronics (ICCAIE)*, 2011, pp. 591–596.

[26] "Oxford Abstracts," https://oxfordabstracts.com/.

[27] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg *et al.*, "Scikit-learn: Machine learning in python," *the Journal of machine Learning research*, vol. 12, pp. 2825–2830, 2011.

[28] S. Bird, "Nltk: the natural language toolkit," in *Proceedings of the COLING/ACL 2006 Interactive Presentation Sessions*, 2006, pp. 69–72.

[29] Y. Vasiliev, *Natural language processing with Python and spaCy: A practical introduction*. No Starch Press, 2020.

[30] S. M. Jain, "Hugging face," in *Introduction to Transformers for NLP: With the Hugging Face Library and Models to Solve Problems*. Springer, 2022, pp. 51–67.

[31] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "BERT: pre-training of deep bidirectional transformers for language understanding," *CoRR*, vol. abs/1810.04805, 2018. [Online]. Available: http://arxiv.org/abs/1810.04805

[32] Z. Liu, F. Jiang, Y. Hu, C. Shi, and P. Fung, "Ner-bert: a pre-trained model for low-resource entity tagging," *arXiv preprint arXiv:2112.00405*, 2021.

[33] "Regular Expressions," https://docs.python.org/3/library/re.html.

[34] "LXML," https://lxml.de/.

[35] "PDFMiner," https://pypi.org/project/pdfminer/.

[36] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz *et al.*, "Huggingface's transformers: State-of-the-art natural language processing," *arXiv preprint arXiv:1910.03771*, 2019.

[37] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga *et al.*, "Pytorch: An imperative style, high-performance deep learning library," *Advances in neural information processing systems*, vol. 32, 2019.

[38] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard *et al.*, "{TensorFlow}: a system for {Large-Scale} machine learning," in *12th USENIX symposium on operating systems design and implementation (OSDI 16)*, 2016, pp. 265–283.

[39] D. Jurafsky and J. H. Martin, "Speech and language processing: An introduction to natural language processing, computational linguistics, and speech recognition."

[40] E. S. Ristad and P. N. Yianilos, "Learning string-edit distance," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 5, pp. 522–532, 1998.

[41] L. Qi and L. Chen, "A linear-chain crf-based learning approach for web opinion mining," in *Web Information Systems Engineering–WISE 2010: 11th International Conference, Hong Kong, China, December 12-14, 2010. Proceedings 11*. Springer, 2010, pp. 128–141.

[42] S. Xu, Y. Li, and Z. Wang, "Bayesian multinomial naïve bayes classifier to text classification," in *Advanced Multimedia and Ubiquitous Engineering: MUE/FutureTech 2017 11*. Springer, 2017, pp. 347–352.

[43] N. Ye, W. Lee, H. Chieu, and D. Wu, "Conditional random fields with high-order features for sequence labeling," *Advances in neural information processing systems*, vol. 22, 2009.

[44] J. C.-W. Lin, Y. Shao, J. Zhang, and U. Yun, "Enhanced sequence labeling based on latent variable conditional random fields," *Neurocomputing*, vol. 403, pp. 431–440, 2020.

[45] T. Wei, J. Qi, S. He, and S. Sun, "Masked conditional random fields for sequence labeling," *arXiv preprint arXiv:2103.10682*, 2021.

[46] Z. Xing, J. Pei, and E. Keogh, "A brief survey on sequence classification," *ACM Sigkdd Explorations Newsletter*, vol. 12, no. 1, pp. 40–48, 2010.

[47] S. Blasiak and H. Rangwala, "A hidden markov model variant for sequence classification," in *Twenty-Second International Joint Conference on Artificial Intelligence*, 2011.

[48] H. Han, E. Manavoglu, C. L. Giles, and H. Zha, "Rule-based word clustering for text classification," in *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, 2003, pp. 445–446.

[49] M. Abbas, K. A. Memon, A. A. Jamali, S. Memon, and A. Ahmed, "Multinomial naive bayes classification model for sentiment analysis," *IJCSNS Int. J. Comput. Sci. Netw. Secur*, vol. 19, no. 3, p. 62, 2019.

[50] G. Singh, B. Kumar, L. Gaur, and A. Tyagi, "Comparison between multinomial and bernoulli naïve bayes for text classification," in *2019 International Conference on Automation, Computational and Technology Management (ICACTM)*. IEEE, 2019, pp. 593–596.

[51] D. Berrar, "Bayes' theorem and naive bayes classifier," *Encyclopedia of bioinformatics and computational biology: ABC of bioinformatics*, vol. 403, p. 412, 2018.

[52] Y. HaCohen-Kerner, D. Miller, and Y. Yigal, "The influence of preprocessing on text classification using a bag-of-words representation," *PloS one*, vol. 15, no. 5, p. e0232525, 2020.

[53] L. N. Cassel, S. Palivela, S. Marepalli, A. Padyala, R. Deep, and S. Terala, "The new acm ccs and a computing ontology," in *Proceedings of the 13th ACM/IEEE-CS Joint Conference on Digital libraries*, 2013, pp. 427–428.

[54] B. Fetahu, A. Fang, O. Rokhlenko, and S. Malmasi, "Gazetteer enhanced named entity recognition for code-mixed web queries," in *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2021, pp. 1677–1681.

[55] T. M. Breuel, "High performance document layout analysis," in *Proceedings of the Symposium on Document Image Understanding Technology*, 2003, pp. 209–218.

[56] "Levenshtein," https://pypi.org/project/Levenshtein/.

[57] L. L. Haak, M. Fenner, L. Paglione, E. Pentz, and H. Ratner, "Orcid: a system to uniquely identify researchers," *Learned publishing*, vol. 25, no. 4, pp. 259–264, 2012.

[58] Y. Yadav and D. B. C. DESAI, "Confsys - an intelligent conference management system," in *Proceedings of the 27th International Database Engineering and Applications Symposium*, ser. IDEAS '23. New York, NY, USA: Association for Computing Machinery, 2023, p. 127–130. [Online]. Available: https://doi.org/10.1145/3589462.3589463