AI-Enabled Academic Conference Management System

Ranjit Reddy Midde, Geetha Seri Lingala, Hemalatha Dubasi,
Gayathri Mangala and Prashanth Kumar Reddy Pullaiahgari
Department of CSE, Srinivasa Ramanujan Institute of Technology, Rotarypuram Village, B K Samudram Mandal, Anantapuramu - 515701, Andhra Pradesh, India

Keywords: Support Vector Machines (SVM), K-Nearest Neighbors (KNN), Random Forest Algorithm.

Abstract: Academic conferences play a vital role in fostering knowledge sharing and collaboration among researchers and experts. However, managing these events involves complex tasks such as paper submission, peer review, scheduling and communication which are often inefficiently handled. Existing methods and systems show significant limitations such as inconsistent reviewer assignments, reliance on manual sorting of papers and limited interaction between authors and reviewers. These issues make the conference process less efficient. Addressing these issues, AI-enabled Academic Conference Management System aims to overcome these challenges by using machine learning techniques such as Support Vector Machines, K-Nearest Neighbors and Random Forest to automatically classify research papers by domain and assign reviewers based on domain expertise. Additionally, it introduces organized feedback mechanisms via email and supports direct author-reviewer collaboration through video conferencing to facilitate the review process. The system reduces manual effort by automating paper classification, ensuring accurate matches, user-friendly interface enables faster workflows and a more efficient review process.

1 INTRODUCTION

Academic conferences are essential platforms for researchers and experts to share knowledge, foster collaborations, and disseminate groundbreaking research. However, managing these conferences is a complex process that involves numerous interdependent tasks, including paper submissions, reviewer assignments, session scheduling, and participant communication. Traditional conference management systems often struggle to address these challenges effectively. Issues such as manual sorting of research papers, inconsistent reviewer assignments, and inadequate communication channels between authors and reviewers lead to inefficiencies, delays, and potential errors. For instance, manual paper review processes are prone to human error, while poorly matched reviewers may result in superficial evaluations, negatively impacting the quality of feedback and the overall conference experience.

To overcome these challenges, this research introduces an AI-enabled Academic Conference Management System designed to optimize and streamline conference workflows. The system leverages advanced machine learning techniques such as Support Vector Machines (SVM), K- Nearest Neighbors (KNN), and Random Forest for automating classification of research papers into specific domains, ensuring precise reviewer assignments based on domain expertise. Additionally, the system integrates innovative features such as structured feedback mechanisms and direct collaboration through video conferencing. These enhancements facilitate seamless communication between authors and reviewers, accelerate the review process, and foster meaningful interactions. By reducing manual effort, improving collaboration and due to a user-friendly interface the proposed system aims to transform academic conference management, addressing inefficiencies and promoting a more organized and effective review process.

2 RELATED WORK

Several systems for managing conferences are developed to streamline various processes like paper submission, review, and feedback.

Midde, R. R., Lingala, G. S., Dubasi, H., Mangala, G. and Pullaiahgari, P. K. R.
AI-Enabled Academic Conference Management System.
DOI: 10.5220/0013927800004919

In *Proceedings of the 1st International Conference on Research and Development in Information, Communication, and Computing Technologies (ICRDICCT'25 2025) - Volume 5*, pages 328-334
ISBN: 978-989-758-777-1

M. Spilker, F. Prinsen, and M. Kalz (2019) analyzed the impact of technology-enhanced academic conferences on professional development, highlighting benefits such as improved accessibility, interactive learning, and global reach, while addressing challenges like equitable access and interaction quality. Similarly, MyConfree, developed by Metta Santiputri Nindy et al. (2018), leverages PHP, CodeIgniter, and MySQL to streamline workflows with features like paper submissions, call-for-paper announcements, and traditional review processes, demonstrating how technology simplifies conference management.

Other systems further emphasize innovation in conference workflows. Cheng Zheng et al. (2008) introduced a collaborative platform with tools for real-time updates, session scheduling, and participant registration to improve coordination. K. Ahmad et al. presented a system tailored for higher education, offering features like author notifications and deadline alerts. OpenConf, introduced by the Zakon Group (2001), simplifies tasks such as peer reviews, attendee registration, and scheduling with its user-friendly interface, reducing administrative burdens and improving the experience for organizers and participants alike.

A. Malinowski and B. Wilamowski, in their work titled *"Paper Collection and Evaluation through the Internet"* (Proceedings of the 27th Annual Conference of the IEEE Industrial Electronics Society, Denver) , addressed limitations in standalone web-based conference management systems. These systems often lack fault tolerance, and the capacity to support distributed users. To overcome these challenges, they proposed a collaborative conference management system that utilizes advanced collaboration technologies to improve fault tolerance, scalability, and user accessibility.

Rigaux Ph., in *"An Iterative Rating Method: Application to Web-based Conference Management"* (Proceedings of the 2004 ACM Symposium on Applied Computing), proposed a method for collecting user preferences or ratings on a large set of items without asking each user to rate each one. The approach relies on an iterative process, where each step, or ballot, involves users rating a sample of items. A collaborative filtering algorithm predicts the missing ratings along with their confidence levels, which are initially set to zero. Subsequent ballots improve the prediction accuracy, and the system administrator determines when to stop the iteration upon reaching a satisfactory level. This method was applied to assign reviewers to papers prior to the review phase in conference management and was implemented in the MYREVIEW web-based system.

A web-based academic conference management system called ConfSys (Huang et al., 2008) is introduced. ConfSys is intended to assist program chairs, general chairs, and program committees in overseeing the operations of scholarly conferences and to offer conference-related services to authors and attendees. It can post papers, assign them to reviewers automatically, let the chairs change the assignment, debate and rate papers, create the program, register for conferences, gather presentation slides, and more. These features facilitate rapid and simple conference management.

Ware M., in *"Online Submission and Peer-Review Systems"* (Learned Publishing, Vol. 18, No. 4, pp. 245-250, 2005), explored the adoption of online systems for managing submissions and peer reviews in academic publishing. The study highlighted the benefits of such systems, including streamlined submission processes, efficient reviewer assignments, and improved communication between authors, reviewers, and editors. Organizing scientific conferences involves managing paper submissions and reviews, which are crucial yet often complicated tasks. Existing systems are complex, with many unused features, and typically rely on hosting services, raising concerns about data security. This study presents the Online Paper Submission System (OPSS) (Rotikan, Reymon. 2016), an online application created to streamline the process of submitting and reviewing papers.

3 PROPOSED SYSTEM

3.1 Workflow of Proposed System

The workflow of AI-Enabled Academic Conference Management System consists of the following major steps: Step 1: Abstract Submission – Authors submit abstracts through a user- friendly web interface, ensuring ease of access. Step 2: Domain Classification – These abstracts are classified into specific domains using advanced machine learning techniques like SVM, KNN, and Random Forest, enabling accurate domain categorization. Step 3: Reviewer Assignment – Domain-specific reviewers are assigned to evaluate the abstracts and provide feedback. Step 4: Full Paper Submission – Abstracts that meet the required standards lead to the submission of full papers for further review. Step 5: Full Paper Evaluation – These full papers undergo a thorough evaluation process, and feedback is sent to the authors via email. Step 6: Collaboration – Video

conferencing tools facilitate discussions between authors and reviewers, fostering better communication and understanding.

Figure 1 depicts the suggested system's structured process. It begins with the submission of an abstract, which is classified into a relevant domain using machine learning algorithms and ends with the paper submission.

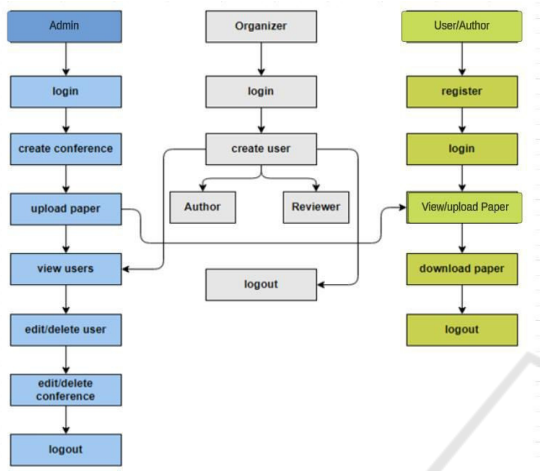

Figure 1: Proposed system workflow.

3.2 Architecture of the Proposed System

Three main roles are included in the centralized database and role-based access control architecture of the AI-Enabled Academic Conference Management System: Admin, Organizer, and User. Admin manages critical functionalities such as creating conferences, uploading research papers, and assigning reviewers to specific domains. Organizers act as intermediaries, handling the creation and management of authors and reviewers while ensuring smooth workflow execution.

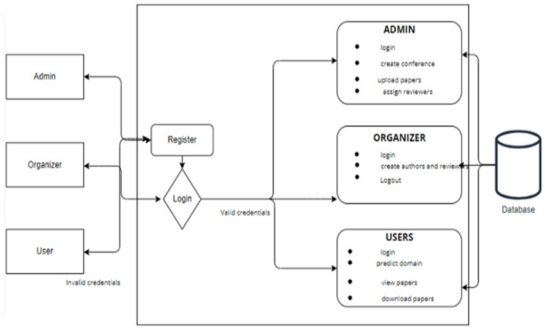

Figure 2: Architecture of proposed system.

Users submit abstracts through a user-friendly web interface, which are then classified into relevant domains using machine learning algorithms like SVM, KNN, and Random Forest. The system ensures efficient handling of data through secure storage and retrieval in the database, with seamless interaction between roles and real-time collaboration enabled via integrated video conferencing tools. This architecture provides a streamlined and scalable framework for managing academic conferences (figure 2).

The system architecture showcases a role-based login mechanism where Admin, Organizer, and Users have distinct functionalities. The Admin manages conferences, uploads papers, and assigns reviewers, while the Organizer handles creating authors and reviewers. Users can log in to predict domains, view, and download papers, with all actions interacting with a centralized database.

4 IMPLEMENTATIONS

4.1 Dataset

Abstracts were manually curated from research papers available online. Each abstract was labeled with its respective domain like cybersecurity, blockchain, cloud computing etc. In the dataset preparation stage, the dataset is loaded from an Excel file, where the first column represents the textual content (abstracts), and the second column contains the corresponding labels. The dataset is examined to identify the number of rows and columns for a clearer understanding of its structure. Text data undergoes preprocessing to ensure uniformity and remove noise. This requires converting text to lowercase, deleting digits and special characters, eliminating stopwords, and performing lemmatization to retrieve the base form of words. After processing, the text input is vectorized to provide numerical feature representations using the TF- IDF (Term Frequency-Inverse Document Frequency) technique. Simultaneously, the labels are encoded into numerical values using a label encoder. Finally, the data is then divided into two parts: 80% for training the model and 20% for testing it. This ensures a fair evaluation of the model's performance.

4.2 Algorithms Used

Three machine learning techniques were implemented for classifying the abstracts into specific domains as follows: Support Vector Machine (SVM) was used to classify research paper abstracts by

creating the best possible boundary in a high-dimensional space. This boundary helps separate the abstracts into their respective domains. The main goal of SVM is to ensure there is as much space as possible between this boundary and the closest data points from each category, which are called support vectors. This approach improves classification accuracy by clearly distinguishing different domains. The SVM classifier was particularly effective for this project because of its ability to handle high-dimensional data, making it ideal for the TF-IDF feature representation of abstracts. Random Forest was used as an ensemble learning algorithm to improve classification performance by constructing multiple decision trees and aggregating their predictions through majority voting. Each decision tree in the forest was trained on different bootstrap sample of dataset, and feature selection was performed randomly at each split, ensuring diversity among the trees. K-Nearest Neighbors (KNN) was employed as a simple, instance-based learning algorithm to classify abstracts by measuring their proximity to labeled examples in the feature space. KNN will operate by calculating the Euclidean distance in between a test sample and all other training samples, selecting k nearest neighbors and determining the class label based on the category that appears most frequently among them.

4.3 Model Training

The Support Vector Machine (SVM), Random Forest, and K-Nearest Neighbors (KNN) models are trained using data that has been split into 80% for training and 20% for testing to ensure reliable evaluation, where 80% of the 330 manually labeled abstracts were used for training and 20% for testing. SVM was tuned to find an optimal hyperplane for separating the data based on their respective domains. Random Forest utilized multiple decision trees, aggregating results through majority voting to improve classification accuracy. KNN classified abstracts based on the proximity of test samples to labeled training examples, using distance-based classification.

4.4 Model Performance Evaluation

The evaluation focuses on accuracy, precision, recall, and F1-score, providing a clear understanding of how well each model performs in classifying abstracts, assigning reviewers, and facilitating the overall workflow. Through these metrics, we aim to validate the efficiency and reliability of the implemented algorithms in real-world conference management scenarios.

4.5 Web Development

A relational database structure to store and manage information, including author submissions, reviewer assignments, paper classifications, review statuses, and video conference schedules. The below Figure 3 depicts ER diagram that illustrates the structure and relationships within the database. It showcases the entities, their attributes, and the connections between them, providing a clear overview of how data is organized and interrelated.

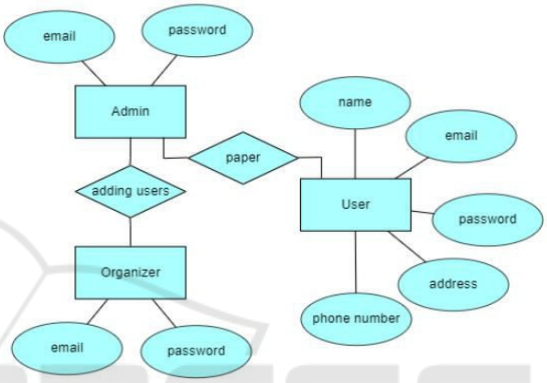

Figure 3: Entities and their relations.

The AI-Enabled Academic Conference Management System uses Django's built-in SQLite database to manage all application tables efficiently. Django's ORM automatically translates models into tables and handles data operations like insertion, updates, and queries. SQLite's lightweight nature and seamless integration with Django make it an ideal choice. The database consists of the following tables: main_author, main_chair, main_conference, main_reviewer, main_user, main_creates, main_paper, main_schedules, main_submits, main_assigns, and main_reviews. The database designed to manage various entities and operations within the system. to the system's functionality. The AI-Enabled Academic Conference Management System was built using Django's MVT (Model-View-Template) architecture, where the Model defines the application's data structure and handles database operations, such as managing entities like authors, reviewers, papers, conferences, and schedules. The View serves as the intermediary, processing user requests, manipulating data through the Model, and passing it to the Template for rendering. The Template is responsible for the user interface, dynamically presenting data using

Django's templating engine, with templates designed for pages like the home page, domain prediction, reviewer assignment, submission status, and schedule, ensuring a user-friendly and visually appealing interface.

4.6 Integration of Machine Learning Model into the Web Page

Machine learning model, trained to classify abstracts into specific domains using algorithms like Support Vector Machines (SVM), Random Forest, and K-Nearest Neighbors (KNN), was seamlessly integrated into the Django-based MVT (Model-View-Template) architecture. After training and evaluating the model using metrics like recall, precision, and F1-score, final model was serialized and saved using the joblib library for efficient reuse in the application. The integration process of the trained machine learning model into the MVT architecture involved several steps. First, the model was serialized and saved as a .joblib file, enabling efficient loading during runtime without the need for retraining. In the View component of the Django framework, the saved model was loaded and utilized for predicting the domain of submitted abstracts. Upon submission, the abstract data was preprocessed and passed to the model for classification. The classification results were then stored in the Model component of the application, associating each abstract with its relevant domain to ensure the appropriate assignment of reviewers based on domain expertise. Finally, the prediction results were dynamically displayed on the Template pages, such as the abstract submission status page and reviewer assignment page, providing users with real-time feedback and facilitating a streamlined workflow.

5 RESULTS

Table 1: Performance ranking and accuracy.

Rank	Algorithm	Precision	Recall	F1-Score	Accuracy
1	Random Forest	84%	85%	84%	89%
2	Support Vector Machine (SVM)	87%	86%	86%	86%
3	K-Nearest Neighbors (KNN)	76%	82%	76%	82%

Performances of different Algorithms were checked and contrast has been made to identify the most efficient algorithm for the domain prediction of the uploaded abstract.

Table 1 represents performance comparison of the machine learning techniques used in the system based on metrics like F1-score, accuracy, precision, and recall. Random Forest algorithm achieved the highest accuracy at 89%, with strong precision and recall values, making it the top- performing model. The Support Vector Machine (SVM) closely followed, demonstrating balanced performance across all metrics. K-Nearest Neighbors (KNN), while showing reasonable recall, had lower precision and F1-scores, ranking third. This ranking highlights the effectiveness of Random Forest for domain classification tasks in the system.

Finally, the Random Forest model is seamlessly integrated with the web interface using Django, making the system highly user-friendly. This integration allows for easy and accurate domain prediction of abstracts. Additionally, other functionalities, such as creating conferences and reviewing abstracts, are designed with intuitive and user-friendly interfaces. The AI-enabled Academic Conference Management System is designed to support four distinct user roles: Admin, Chair/Organizer, Author, and Reviewer. Each role is provided with a tailored interface that allows them to perform specific tasks relevant to their responsibilities in the conference management process. The following key web pages illustrate how the interface accommodates the needs of each role.

Figure 4: Home page.

Figure 4 depicts homepage that serves as a central

hub, offering intuitive navigation tailored to authors, reviewers, and administrators. It streamlines access to role-specific features, enabling authors to submit papers, reviewers to evaluate assignments, and administrators to manage workflows efficiently, ensuring a seamless user experience.

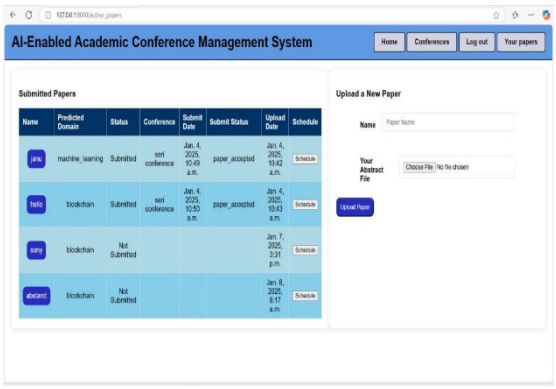

Figure 5: Page for domain prediction.

Figure 5 depicts the abstract classification page, where page highlights the predicted domain and provides transparency into the classification results. The machine learning model is integrated into the webpage using Django, enabling seamless interaction between the backend and the user interface. All the conference details are stored in the backend using sqlite database which is a built in database in Django.

Figure 6: Page for reviewer assignment.

Figure 6 depicts the reviewer assignment page, where administrators assign reviewers to abstracts based on their classified domains. The interface displays the abstract details, predicted domain, and a list of available reviewers with their respective expertise. This ensures that reviewers are matched with abstracts relevant to their domain, improving the quality of feedback and reducing manual workload. The streamlined process enhances the efficiency of the review workflow.

6 FUTURE SCOPE

The future scope of the AI-enabled Academic Conference Management System includes integrating multiple conferences on a unified platform, enabling centralized management and resource sharing. The system can be enhanced with scalable conference management, dynamic reviewer matching across events, and real-time collaboration tools for authors and reviewers. Mobile application development can improve global accessibility, while AI-driven analytics can offer valuable insights for organizers. Additionally, integrating blockchain technology can enhance transparency, and automated session scheduling can streamline event organization, making the platform a comprehensive solution for academic conferences.

7 CONCLUSIONS

The AI-enabled Academic Conference Management System successfully achieves its objectives of automating abstract classification, streamlining reviewer assignment, and enhancing collaboration between authors and reviewers. By employing machine learning techniques like Random Forest, K-Nearest Neighbors, and Support Vector Machines, the system ensures precise domain classification of abstracts, facilitating efficient reviewer allocation. Integrated tools, including structured feedback mechanisms and video conferencing, further promote effective communication and collaboration. The platform addresses inefficiencies in traditional conference workflows by providing an organized, transparent, and user-friendly interface for all stakeholders. With these advancements, the system establishes a strong foundation for modernizing academic conferences while meeting the objectives set forth in the project.

REFERENCES

Cheng Zheng, Weiming Shen, Qinghua Zheng, Feng Tian, "Design and Implementation of a Collaborative Conference Management System", 2008.

Huang, M., Y. Feng, B. C. Desai. CONFSYS: A Webbased Academic Conference Management System. Proceedings of 2008 C3S2E conference. ACM Press, 2008.

K. Ahmad, A. A. Abdullah, and A. M. Zeki, “Web-based Conference Management System for Higher Learning Institutions,” International Islamic University Malaysi

a, Kulliyyah of Information and Communication Technology.

M. Spilker, F. Prinsen and M. Kalz, "Valuing technology-enhanced academic conferences for continuing professional development", A systematic literature review Professional Development in Education, 2019.

Malinowski and B. Wilamowski, "Paper Collection and Evaluation through the Internet", Proceedings of the 27 Annual Conference of the IEEE Industrial Electronics Society, Denver.

Metta Santiputri, Nindy, Surya Agustin, MeraKartika Delimayanti, “MyConfree: A Web- based Conference Management System” Published by IEEE, Proceedings of IEEE 2018.

Rigaux Ph. An Iterative Rating Method: Application to Web-based Conference Management. Proceedings of the 2004 ACM symposium on applied computing.

Rotikan, Reymon. 2016. OPSS: Online Paper Subm-ission System Untuk Kegiatan Konferensi Ilmiah Cogito Smart Journal/VOL.2/No/2/Desember 2016.

Ware M. “Online submission and peer-review systems” in learned publishing.

Zakon Group, “OpenConf Conference Management System,” Web Site: www.openconf.org, 2001.