CS19643-FOUNDATIONS OF MACHINE LEARNING

URCAREER: JOB RECOMMENDATION SYSTEM

URCAREER

BHARATHEESHWAR S (2116210701041)
GOKULA KRISHNA H (2116210701062)

PROJECT COORDINATOR:
Dr.S.Vinod Kumar
Professor
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

RAJALAKSHMI ENGINEERING COLLEGE

Abstract

The "urcareer: Job Recommendation System" presents a groundbreaking approach to revolutionize the job search experience through the integration of advanced machine learning (ML) algorithms. Traditional job search platforms typically rely on rudimentary keyword matching and simplistic filters, which may not adequately capture the nuanced preferences and skills of individual users.The "urcareer" system addresses these challenges by leveraging a hybrid approach that combines content-based and collaborative filtering ML algorithms. Content-based filtering utilizes natural language processing (NLP) techniques to analyze job descriptions and user profiles, ensuring that recommended jobs closely align with the skills, qualifications, and preferences of each user. Concurrently, collaborative filtering harnesses user interaction data to identify similarities between users and recommend jobs that have been positively received by users with comparable profiles..In summary, the "urcareer: Job Recommendation System" represents a significant advancement in job search .

Problem Statement

In today's rapidly evolving job market, the process of finding suitable employment opportunities is often fraught with challenges and inefficiencies. Traditional job search methods rely heavily on manual efforts and generic algorithms, resulting in a lack of personalized recommendations and suboptimal matches between job seekers and available positions. Moreover, existing job search platforms often struggle to adapt to the diverse needs and preferences of individual users, leading to frustration and dissatisfaction among job seekers.The "urcareer: Job Recommendation System" seeks to address these shortcomings by leveraging advanced machine learning (ML) algorithms to deliver highly personalized job recommendations tailored to each user's unique skills, qualifications, and preferences. The primary challenge lies in developing a robust recommendation engine that can effectively analyze and interpret complex job descriptions and user profiles, extracting meaningful insights to facilitate accurate matching.Additionally, the system must overcome the inherent limitations of traditional recommendation approaches by integrating a hybrid model that combines content-based and collaborative filtering techniques. This hybrid approach presents its own set of challenges, including algorithmic complexity, data integration, and scalability issues.

Existing System

In the current landscape of job search platforms, the process of finding suitable employment opportunities is often characterized by inefficiencies and limitations. Traditional job search platforms primarily rely on basic keyword matching and simplistic filters, which may lead to irrelevant or inadequate job recommendations for users. These platforms often lack the ability to provide personalized recommendations that align closely with the skills, qualifications, and preferences of individual users. Additionally, they may not effectively leverage user interaction data to improve the relevance and accuracy of job suggestions over time.Moreover, existing job search systems typically lack advanced machine learning (ML) capabilities, which are essential for delivering highly personalized recommendations. Without ML algorithms, these systems struggle to analyze and interpret complex job descriptions and user profiles, resulting in suboptimal matching between job seekers and job opportunities.As such, there is a pressing need for a more sophisticated and efficient job recommendation system.Overall, the existing job search landscape is characterized by its limitations in providing personalized and accurate job recommendations.

Proposed System

The proposed "urcareer: Job Recommendation System" aims to revolutionize the job search process by introducing an advanced platform that harnesses the power of machine learning (ML) algorithms. Unlike traditional job search platforms that often provide generic and impersonalized recommendations, the proposed system will offer highly tailored job suggestions based on the unique skills, preferences, and career objectives of each user.

At the core of the proposed system lies a hybrid approach that integrates content-based and collaborative filtering ML algorithms. Content-based filtering utilizes natural language processing (NLP) techniques to analyze job descriptions and user profiles, enabling the system to identify relevant job opportunities that closely match the user's qualifications and interests. Meanwhile, collaborative filtering leverages user interaction data to identify patterns and similarities between users, allowing the system to recommend jobs that have been positively received by users with similar profiles.

Aim and Objectives

Aim:

The aim of the "urcareer: Job Recommendation System" is to provide a comprehensive and personalized solution to enhance the job search process for users. The primary objective is to develop a sophisticated recommendation system that leverages machine learning algorithms to deliver tailored job suggestions based on users' skills, qualifications, and preferences.

Objectives:

1. Implementing a hybrid approach combining content-based and collaborative filtering algorithms to ensure highly relevant job recommendations.

2. Integrating natural language processing (NLP) techniques to analyze job descriptions and user profiles effectively.

Literature Survey

The field of career guidance and job recommendation has evolved significantly in recent years, driven by advancements in technology and a growing demand for personalized and data-driven solutions. Historically, career counseling has relied on traditional models such as trait-factor theory and Holland's theory of vocational choice. These models focus on matching individuals with careers based on personality traits, interests, and aptitudes. While these approaches have provided valuable insights into career decision-making processes, they often lack the flexibility and customization needed to address the diverse needs of today's workforce.Recent research has highlighted the potential of technology to enhance career guidance services. Tools such as online career assessments, virtual career fairs, and AI-powered chatbots have emerged as effective means of delivering personalized career advice and resources to individuals. For example, studies by Savickas (2019) and Sampson et al. (2020) emphasize the importance of integrating technology into career counseling to reach a wider audience and provide scalable solutions.

Literature Survey

The use of natural language processing (NLP) techniques for resume analysis has gained traction in the field of human resources and career guidance. Research by Gugnani et al. (2018) demonstrates the effectiveness of NLP in extracting key information from resumes, such as skills, experiences, and qualifications. By automating the resume screening process, NLP algorithms can save time and resources for both job seekers and employers.Machine learning (ML) algorithms offer promising opportunities for assessing individual skills and competencies in a data-driven manner. Studies by AlZoubi et al. (2021) and Sina et al. (2020) explore the use of ML models for predicting job performance and identifying skill gaps. Personalized job recommendation systems leverage user data and machine learning algorithms to match individuals with relevant job opportunities. Research by Li et al. (2019) and Zhou et al. (2021) demonstrates the effectiveness of recommendation algorithms in improving job search outcomes and user satisfaction.

Hardware & Software Requirements

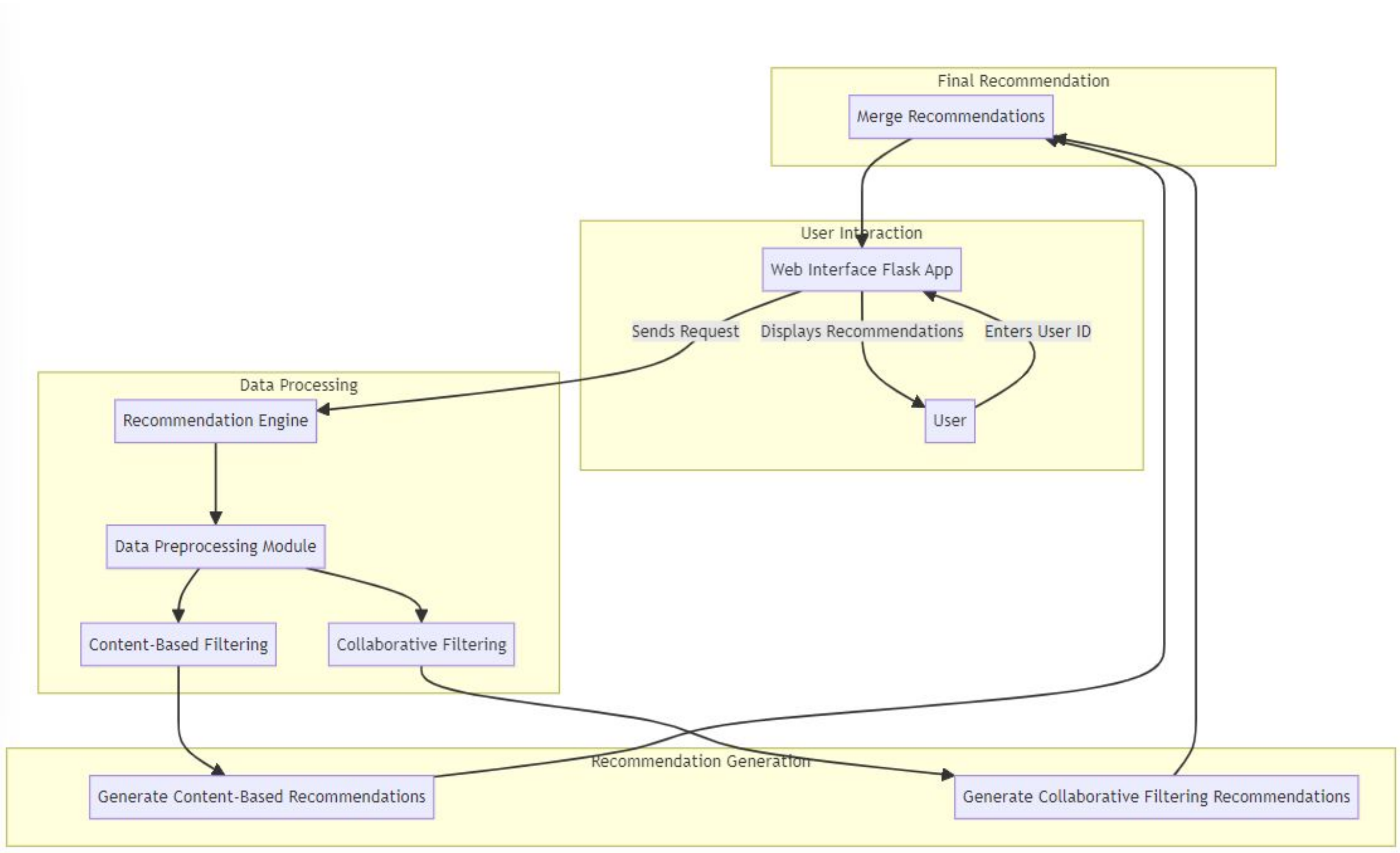
Hardware Requirements:

- Processor: Intel Core i3 or equivalent
- RAM: 4GB or higher
- Storage: 128GB SSD or higher
- Display: 15-inch monitor or larger

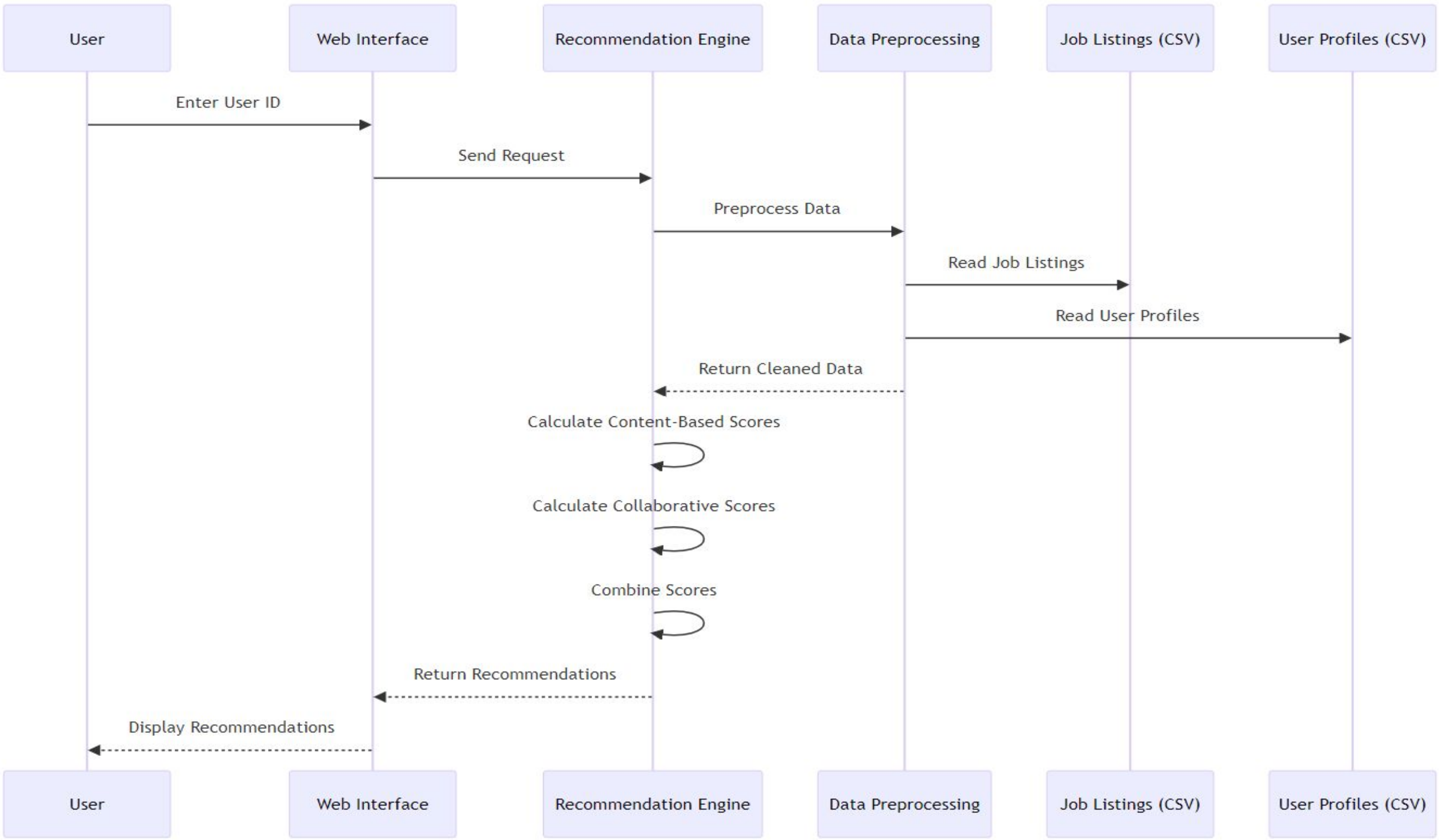
Software Requirements:

- Python 3.8 or higher
- Flask Framework
- Pandas Library
- Scikit-learn Library
- Spacy Library
- HTML/CSS/JS for Frontend Development

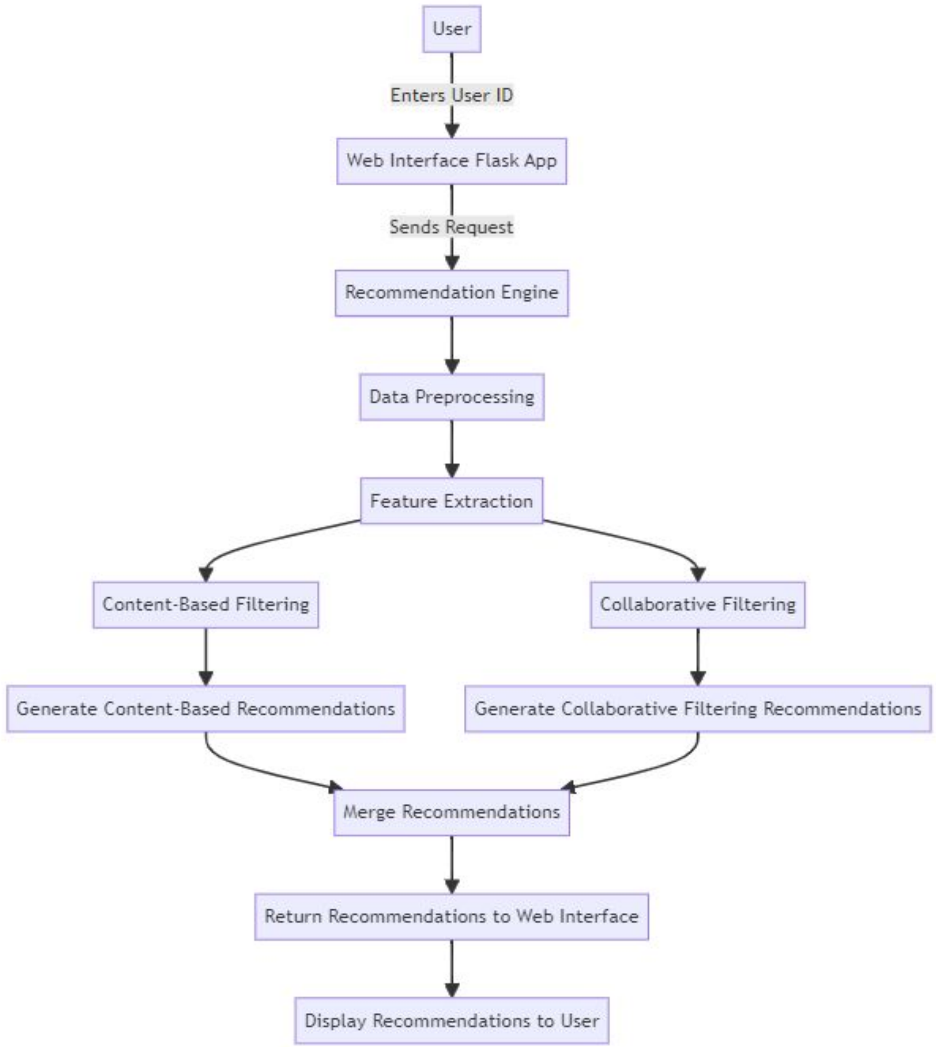
Architecture Diagram



SEQUENCE DIAGRAM



Flow Diagram



Functional Description

MODULES

1. Data Preprocessing Module:

This module is responsible for processing and cleaning the raw data obtained from job listings and user profiles. It utilizes natural language processing (NLP) techniques through libraries such as spaCy to tokenize, lemmatize, and remove stop words from job descriptions. Additionally, it implements data cleaning operations to handle missing values, incorrect formats, and outliers in the dataset. Libraries like pandas and NumPy are employed for efficient data manipulation and preprocessing operations.

2. Feature Extraction Module:

The Feature Extraction Module extracts relevant features from the preprocessed data to represent job descriptions and user profiles. It utilizes techniques such as TF-IDF (Term Frequency-Inverse Document Frequency) vectorization to convert textual data into numerical feature vectors. The module also employs libraries like scikit-learn for implementing feature extraction algorithms and generating feature matrices.

Functional Description

3. Content-Based Filtering Module:

This module implements content-based recommendation algorithms to suggest jobs based on the similarity between job descriptions and user profiles. It utilizes cosine similarity or other distance metrics to measure the similarity between feature vectors representing job descriptions and user profiles.The module relies on libraries like scikit-learn for implementing similarity calculations and integrating NLP models.By focusing on the content of the jobs, including skills required, job titles, and descriptions, this module excels at suggesting relevant opportunities that align with the user's expertise and interests.

4. Collaborative Filtering Module:

The Collaborative Filtering Module implements collaborative filtering algorithms to recommend jobs based on user interaction data and similarities between users. It utilizes user-item matrices to capture user preferences and interactions with job listings..Techniques such as matrix factorization, including Singular Value Decomposition (SVD) or Alternating Least Squares (ALS), are employed to uncover latent features representing user preferences and job characteristics.

Functional Description

5. Hybrid Recommendation Module:

The Hybrid Recommendation Module integrates content-based and collaborative filtering approaches to provide enhanced job recommendations. It combines the strengths of both approaches to mitigate their respective weaknesses and provide more accurate and diverse recommendations. Ensemble techniques or weighted combination strategies are implemented to fuse recommendation scores from content-based and collaborative filtering algorithms. The module utilizes libraries like NumPy for efficient array manipulation and ensemble modeling.

6. User Interface Module:

The User Interface Module develops a user-friendly interface for users to interact with the recommendation system. It implements web-based or desktop GUIs using frameworks like Flask or Django for seamless integration with backend modules. HTML, CSS, and JavaScript are utilized for designing responsive and visually appealing user interfaces. Interactive elements and real-time updates are incorporated to enhance user experience and engagement.

Sample coding

urcareer.py

```
app = Flask(__name__)
CORS(app)
def preprocess(text):
    doc = nlp(text)
    tokens = [token.lemma_ for token in doc if not token.is_stop and token.is_alpha]
    return ' '.join(tokens)
job_data['processed_description'] = job_data['description'].apply(preprocess)
vectorizer = TfidfVectorizer(max_features=1000)
tfidf_matrix = vectorizer.fit_transform(job_data['processed_description'])
similarity_matrix = cosine_similarity(tfidf_matrix, tfidf_matrix)
user_item_matrix = np.random.rand(user_data.shape[0], job_data.shape[0])
def hybrid_recommendations(user_id):
    content_scores = similarity_matrix.dot(user_item_matrix[user_id])
    collaborative_scores = user_item_matrix[user_id]
    combined_scores = 0.5 * content_scores + 0.5 * collaborative_scores
    return np.argsort(combined_scores)[::-1][:3]
if __name__ == '__main__':
    app.run(debug=True)
```

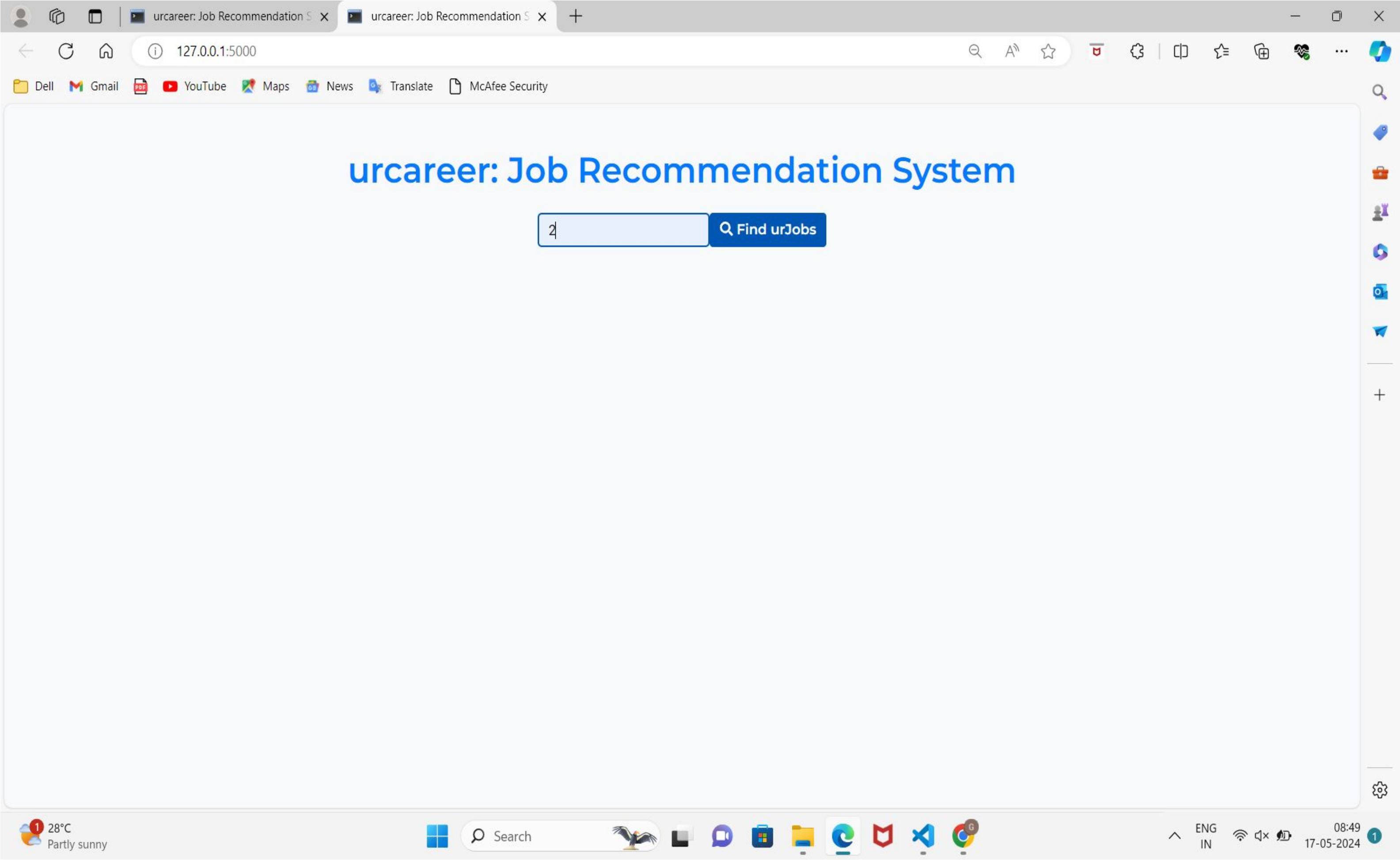
Sample coding

urcareerapp.ipynb

```
def get_collaborative_recommendations(user_id, user_similarity, top_n=3):
    similar_users = list(enumerate(user_similarity[user_id]))
    similar_users = sorted(similar_users, key=lambda x: x[1], reverse=True)[1:top_n+1]
    recommended_jobs = set()
    for user, score in similar_users:
        user_jobs = set(interaction_df[interaction_df['Resume_ID'] == user]['Job_ID'])
        recommended_jobs = recommended_jobs.union(user_jobs)
    return recommended_jobs


def get_content_based_recommendations(resume_id, cosine_similarities, top_n=3):
    sim_scores = list(enumerate(cosine_similarities[resume_id]))
    sim_scores = sorted(sim_scores, key=lambda x: x[1], reverse=True)[:top_n]
    recommended_jobs = [job[0] for job in sim_scores]
    return recommended_jobs
content_recommendations = get_content_based_recommendations(0, cosine_similarities)
print("Content-Based Filtering Recommendations for Resume 0:", content_recommendations)
```

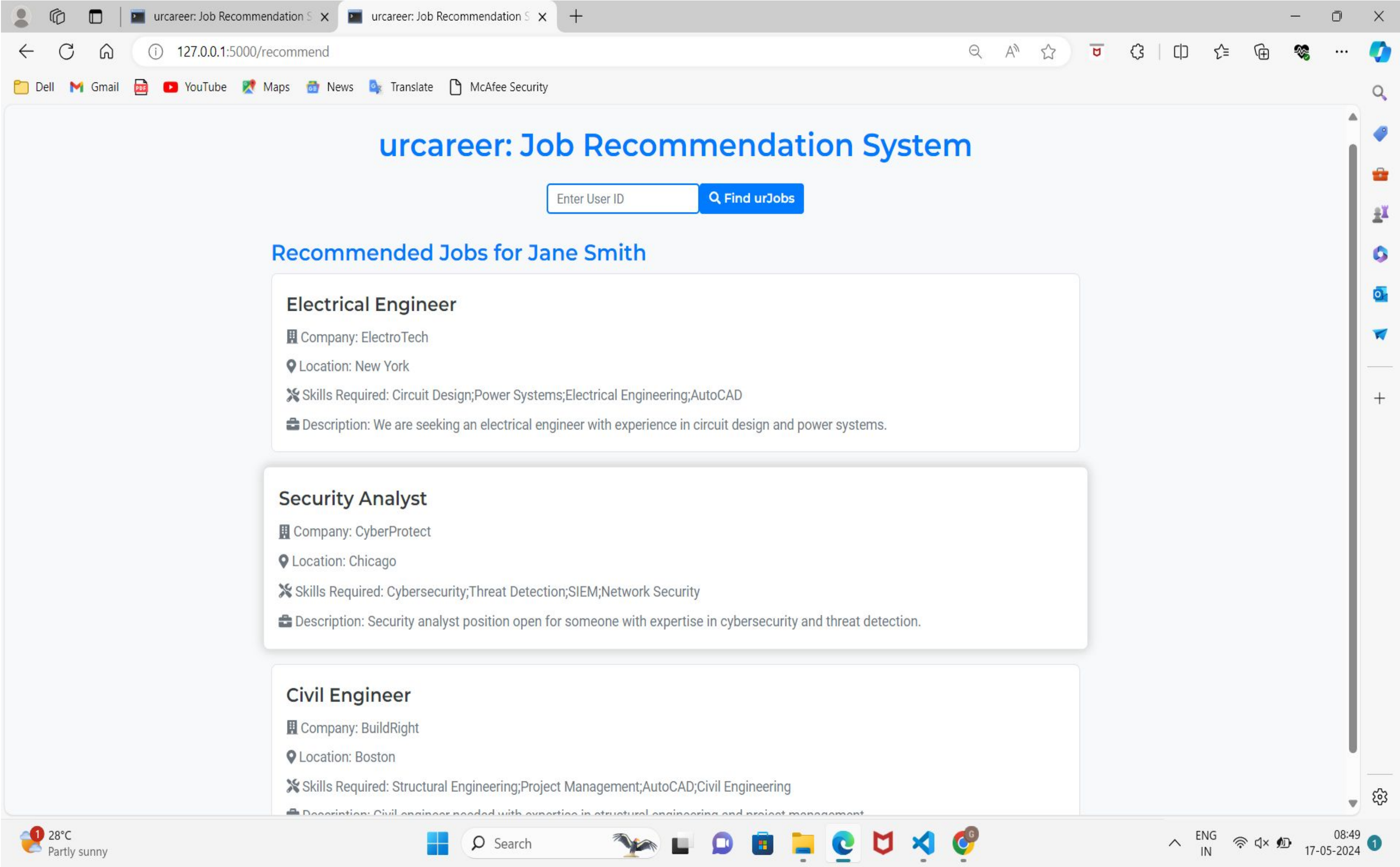
Outputs and Final Results

urcareer Front-end: Filter by user-id



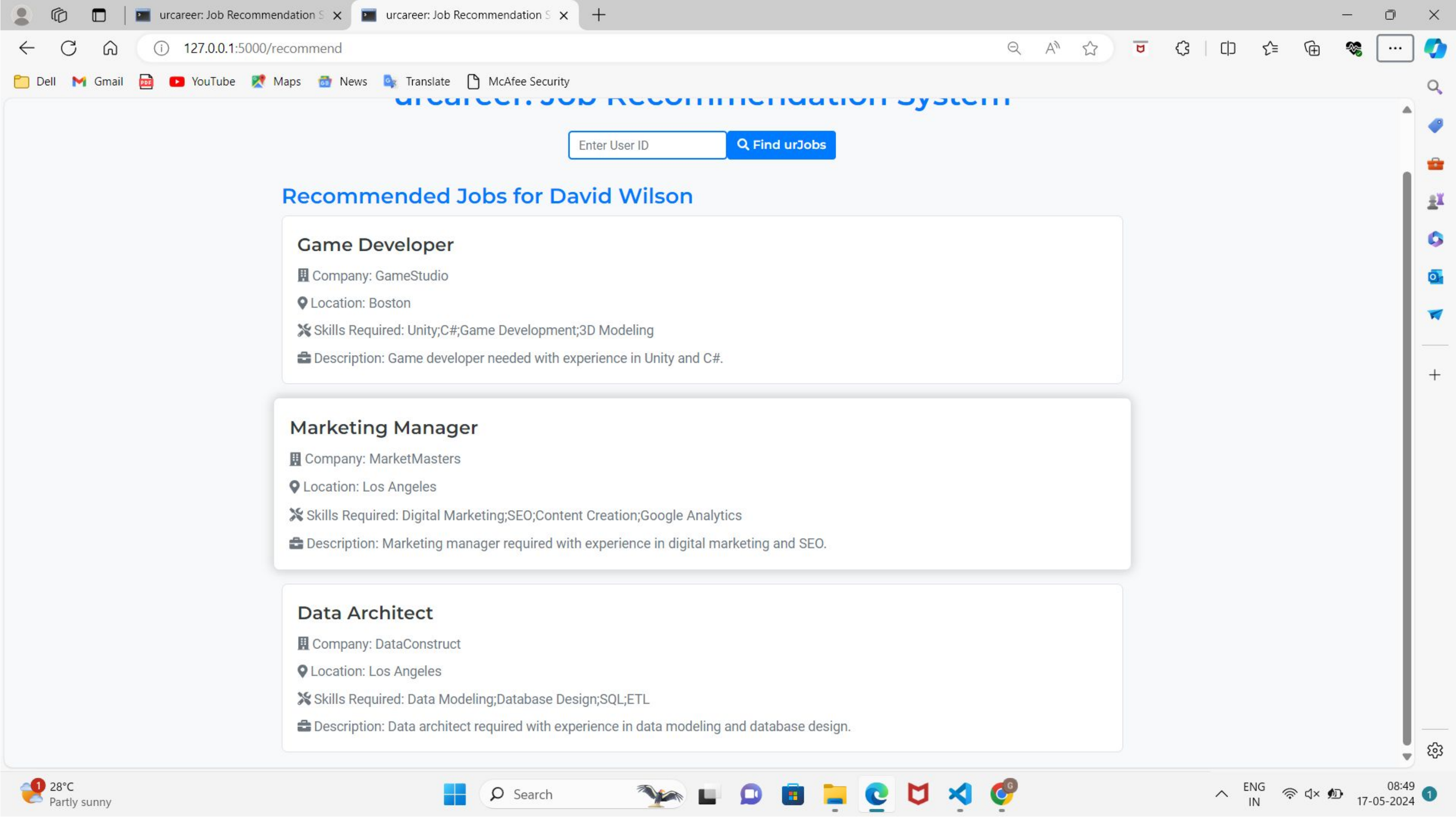
Outputs and Final Results

urcareer Front-end: Job recommendation for User 1



Outputs and Final Results

urcareer Front-end: Job recommendation for User 2



Outputs and Final Results

Dataset used: job_listings.csv,user_profiles.csv

```
jobs_df = pd.read_csv('job_listings.csv')
jobs_df.head()
```

	id	title	description	company	location	skills
0	1	Data Scientist	We are looking for a data scientist with exper...	TechCorp	New York	Python;R;Machine Learning;Statistics
1	2	Software Engineer	Join our team as a software engineer. Must be ...	Innovatech	San Francisco	Java;Spring;Microservices;Docker
2	3	Web Developer	Seeking a web developer skilled in HTML, CSS, ...	WebWorks	Los Angeles	HTML;CSS;JavaScript;React
3	4	Data Analyst	Data analyst position available. Must know SQL...	DataSolutions	Chicago	SQL;Tableau;Excel;Data Analysis
4	5	DevOps Engineer	DevOps engineer needed with experience in AWS,...	CloudNet	Seattle	AWS;Jenkins;Kubernetes;CI/CD

```
resume_df = pd.read_csv('user_profiles.csv')
resume_df.head()
```

	user_id	name	experience_years	skills	location
0	1	John Doe	5	Python;Machine Learning;Data Analysis;R	New York
1	2	Jane Smith	3	Java;Spring;Microservices;Docker	San Francisco
2	3	Bob Johnson	7	HTML;CSS;JavaScript;React	Los Angeles
3	4	Alice Williams	2	SQL;Tableau;Excel;Data Analysis	Chicago
4	5	Michael Brown	10	AWS;Jenkins;Kubernetes;CI/CD	Seattle

Outputs and Final Results

urcareer Jupiter Notebook: Content-based Filtering

urcareer

C:/Users/Krishna/Downloads/urcareer%20(1).html

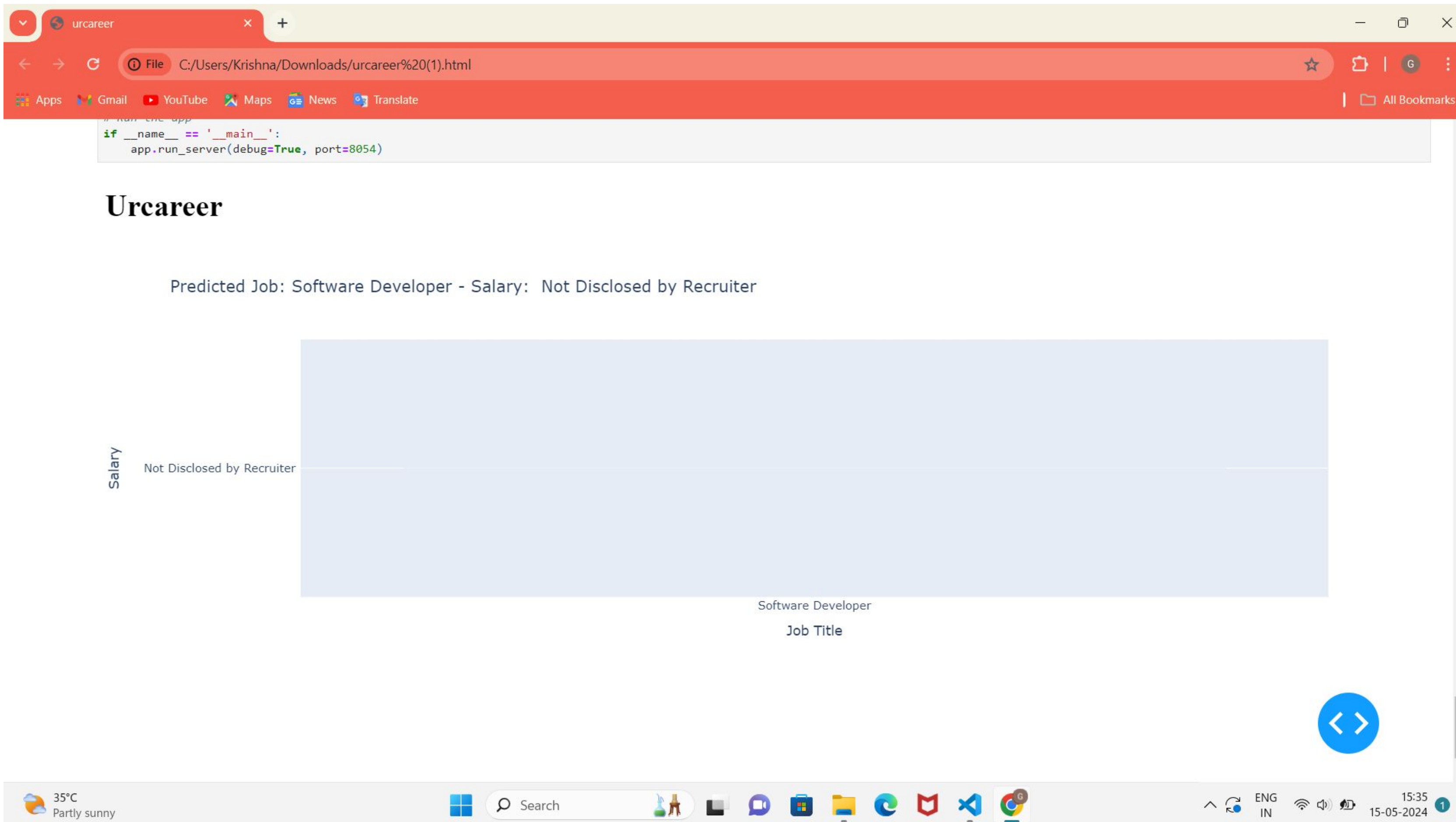
```
                          align='center',
                          font=dict(size=14, color='black')),
                   cells=dict(values=[recommended_jobs_df[col] for col in recommended_jobs_df.columns],
                              fill_color='white',
                              align='center',
                              font=dict(size=12, color='black')))
               ],
               'layout': dict(height=500, margin=dict(l=20, r=20, t=30, b=20))
           }
       )
   ])
])

# Run the app
if __name__ == '__main__':
    app.run_server(debug=True,port=8052)
```

Job Recommendations Based on Skills

Job Title	Job Experience Required	Key Skills	Role Category	Functional Area	Industry
Software Developer	6 - 9 yrs	Power Bi| Data Modeling| Database Development| SQL	Programming & Design	IT Software - Other	IT-Software, Software Services
Software Developer	4 - 6 yrs	cloud| Azure| Power Bi| Bi| SSRS| ms sql server| SQL Server| SSIS	Programming & Design	IT Software - Application Programming , Maintenance	IT-Software, Software Services
Software Developer	1 - 2 yrs	Business Intelligence| Power BI| MSBI| SQL Server| Analytics| SQL| T - SQL	Programming & Design	IT Software - DBA , Datawarehousing	IT-Software, Software Services

Outputs and Final Results

urcareer Jupiter Notebook: Content-based Filtering



Conclusion and Future Enhancements

In conclusion, the urcareer: Job Recommendation System presents a comprehensive solution for assisting users in their job search endeavors. Through the integration of advanced machine learning algorithms, including hybrid content-based and collaborative filtering techniques, the system delivers personalized job recommendations tailored to individual preferences and expertise. The project successfully addresses the challenge of information overload in the job market by providing users with relevant and targeted job suggestions based on their profiles and interactions.Looking ahead, several avenues for future enhancements and expansions exist to further enrich the capabilities of the system. Firstly, the integration of additional data sources, such as social media profiles or professional networks, could enhance the system's understanding of user preferences and improve recommendation accuracy. Furthermore, incorporating contextual information, such as industry trends or geographical considerations, could enable the system to offer more contextualized recommendations that align with evolving user needs and market dynamics.

References

- Stephen Marsland, “Machine Learning – An Algorithmic Perspective”, Second Edition, Chapman and Hall/CRC Machine Learning and Pattern Recognition Series, 2014.
- Peter Flach, “Machine Learning: The Art and Science of Algorithms that Make Sense of Data‖”, First Edition,Cambridge University Press, 2012.
- Tom M Mitchell, “Machine Learning”, First Edition, McGraw Hill Education, 2013.
- Trevor Hastie, Robert Tibshirani and Jerome Friedman, “The Elements of Statistical Learning (ESL)”, 2nd edition,Springer, 2016. ISBN 978-0387848570.
- S. Russell and P. Norvig, Artificial Intelligence: A Modern Approach, Prentice Hall, Third Edition, 2015.
- Nils J. Nilsson, Artificial Intelligence: A New Synthesis (1 ed.), Morgan-Kaufmann, 1998. ISBN 978-1558605350.

Thank You

URCAREER