Intelligent Job Search: A Data-Driven Method

Sanjay M

School of computer

Science and

Engineering

VIT Chennai, India

sanjay.m2020@Vitstudent.ac.in

Sriganth R

School of computer

Science and

Engineering

VIT Chennai, India

Sriganth.r2020@vitstudent.ac.in

***Abstract*— Studies indicate that when it comes to work and professional development, finding the right opportunity might be difficult. To address this issue, a data-driven approach is developed that completely changes the way in which job searchers choose their career routes. Through the application of large-scale data analytics and machine learning techniques, our study pioneers improved job suggestions. This study offers a data-driven strategy to provide job searchers with more assistance on recommended jobs. Our technology uses big data analytics and machine learning to examine user profiles, skill sets, and market trends in order to provide tailored job suggestions. It increases the efficacy and efficiency of job searches by matching user preferences with job criteria. Furthermore, the system is always improving its recommendations as it adjusts to changing market circumstances.**

***Keywords— professional development, job, user, suggestions, study, job searchers, searchers ,order, machine learning***

1. INTRODUCTION

The way society has evolved in the digital era has had a major impact on the constantly shifting job market in recent years. Advanced recommendation algorithms are desperately needed, as seen by the proliferation of online employment marketplaces and the wide range of career alternatives available. This research presents a large dataset of various job advertisements from various industries, which is a helpful resource to understand the complex employment landscape. Our project focuses on utilising this dataset to create a novel recommendation engine that goes beyond traditional approaches. In addition to empowering job seekers, this approach is intended to help labour market analysts and researchers. Important dataset variables such as "Job Title," "Skills," "Education," "Experience," "Industry," and "Pay Rate" form the basis of our recommendation engine.

Identifiers like "Id" improve the speed of data referencing, and the "Company" field helps discover possible employers. The "Education" and "Experience" requirements provide important insights into how the labour market is changing. Comprehensive "Job Descriptions" provide candidates with nuanced viewpoints, and the "Job Location Address" helps assess the ramifications of relocation and commute. The careful interpretation of job-specific "Skills," which enables applicants to match their competencies with industry requirements, is at the heart of our recommendation system. The "Industry" classification of businesses helps job seekers align their careers. Additionally, the "Pay Rate" box guarantees clear expectations for remuneration, allowing for informed decision-making. Temporal elements are resolved by the "Post Date," which provides insight into how recent job openings are. By indicating the necessary years, inclusive qualities like "Experience\_upper" and "Experience\_lower" improve understanding.

This suggestion system makes a substantial contribution to professional assistance and a broader examination of the labour market, going beyond the development of specific jobs. By utilising the extensive dataset, our methodology aims to transform job recommendation systems and promote well-informed decision-making within the ever-changing work landscape.

1. LITERATURE SURVEY

In recent years, there has been a lot of interest in developing intelligent career advice systems that use data-driven approaches to help job seekers make informed decisions about their future pathways. This survey investigates existing research and contributions in the topic of Intelligent Career Guidance, emphasising major findings from pertinent studies. This system is made up of three major modules: skill evaluation, prediction, and result analysis [1]. It has exhibited improved accuracy compared to previous systems when using the K Nearest Neighbour algorithm, suggesting a possible path for intelligent career recommendations. Furthermore, the development of Personalised Career-path Recommender Systems (PCRS) has shown significant promise in assisting students, particularly those in engineering disciplines. PCRS uses fuzzy intelligence to tailor assistance depending on academic performance, personality type, and extracurricular activities [2]. This customised approach to specific communities and regions fills a vital vacuum in career advising giving. Furthermore, literature evaluations on job recommender systems have defined the architecture, implementation, and approaches aimed at improving accuracy and speed. They emphasise the importance of user profiles and recommendation technologies that incorporate hybrid user profiles and relevant feedback [3]. Furthermore, the use of reciprocal recommender technology has been investigated, showing the possibility for future study areas in the subject. Furthermore, in the literature, career recommendation systems using content-based filtering, machine learning, and natural language processing approaches have been discussed [4]. These systems collect user preferences and abilities, use machine learning to personalise job possibilities, and employ natural language processing to analyse user comments, resulting in more exact career suggestions. The study on recommender systems in the realm of social media resources, emphasising their usage of user-generated content, is also reviewed in the literature. Notably, one strategy discussed in the literature uses machine learning algorithms to assess students' abilities and provide personalised job recommendations. This system is made up of three main modules: skill evaluation, prediction, and result analysis [1]. It has showed improved accuracy compared to earlier systems when using the K Nearest Neighbour algorithm, providing a viable approach for intelligent career recommendations. Furthermore, the development of Personalised Career Path Recommender Systems (PCRS) has demonstrated considerable promise in guiding students, particularly in engineering disciplines. PCRS employs fuzzy intelligence to tailor assistance depending on academic success, personality type, and extracurricular abilities [2]. This targeted approach to individual communities and areas fills a critical hole in career advising. Furthermore, evaluations of the literature on job recommender systems have examined the structure, implementation, and approaches used to improve accuracy and performance. They emphasise the significance of user profiles and recommendation systems, such as hybrid user profiles and relevance feedback [3]. This demonstrates the expansion of study in this field and opens the door to future discoveries. Furthermore, data mining techniques have been applied to job recommender systems in order to provide personalised job recommendations based on user profiles and preferences [7]. This method uses classification rules and content-based matching to categorise candidate and job data, beating typical recommender systems in prediction accuracy. A comparison of user-based and item-based collaborative filtering algorithms revealed information about their performance in job recommendation systems [8]. Item-based collaborative filtering outperforms other methods, especially when using the Log likelihood similarity technique. Finally, a hybrid recommender system using Bayesian networks that combines content-based and collaborative filtering strategies has been presented [9]. This model outperformed other hybrid, collaborative, and content-based models in terms of accuracy. In summary, the literature review sheds light on the various techniques and methodologies used in the subject of Intelligent Career Guidance. These research help to enhance data-driven approaches that enable job searchers to make well-informed career decisions [10].

*2.1 OBJECTIVE*

This paper endeavors to develop a robust job recommendation system by harnessing the wealth of information within the provided postings dataset. It emphasizes key attributes, including "Job Title," "Skills," "Education," "Experience," "Industry," and "Pay Rate." The primary objective is to empower job seekers with data-driven insights, facilitating an enhanced job search experience by aligning their qualifications and preferences with suitable job postings. Utilizing this comprehensive dataset, our system aims to streamline candidate sourcing and selection processes for job recruiters. The goal is to efficiently identify and connect with potential candidates who best fit the job requirements, ultimately improving recruitment effectiveness and reducing time-to-hire.

Acknowledging the dynamic nature of the job market, a crucial aspect of our recommendation system is prioritizing recent job postings. This is accomplished by giving careful consideration to the "Post Date" attribute, ensuring that recommendations remain relevant and up-to-date for users actively seeking employment. Beyond the core functionality of job recommendations, our system incorporates additional features to enhance user experience further. These encompass skill gap analyses, providing valuable insights for job seekers to identify areas for skill improvement, industry trend analysis based on the post date to offer recruiters real-time market insights, and skill tests to assess and validate candidates' proficiency in specific competencies. This comprehensive approach aims to revolutionize not only the job recommendation landscape but also provide a platform catering to diverse needs.

*2.2 DATASET DESCRIPTION*

• **Id:** A unique identification issued to each job posting entry to make referencing and data administration easier.

• **Company:** The name of the company that posted the position, which allows users to discover possible employers.

• **Education:** Specifies the educational qualifications required for the specified employment vacancy, giving applicants an idea of the degree of education expected.

• **Experience:** The quantity of work experience required for candidates to be considered for the position.

•  **Job Description:** A full summary of the job posting that includes responsibilities, credentials, and other relevant information to help job seekers understand the role.

• **Job Location Address:** Indicates the geographic location of the job.

based, assisting candidates in weighing commuting and relocation options.

• **Job Title**: The job posting's title or designation, which provides job seekers with an initial grasp of the role.

• **Pay Rate**: Indicates the pay or range of compensation linked with the job posting, allowing candidates to determine remuneration expectations.

• **Post Date**: The date the job posting was made, which is important for determining the freshness of possibilities.

• **skills:** A list of required abilities and competencies for the job, providing vital information for applicants to assess their qualifications.

• Industry: Indicates the industry or sector to which the company belongs, which assists job seekers in aligning their careers with specific fields.

• **Experience\_upper:** Basically, it has an experience range, which contains information about the required years of experience for the position.

• **Experience\_lower:** Information about the required years of experience for the job.

1. **PROPOSED WORK**

Drawing insights from the details provided in this research, a dataset underwent collection, and the initial steps of preprocessing were initiated. These pivotal preprocessing stages encompassed the utilization of diverse Natural Language Processing (NLP) techniques. Additionally, the dataset underwent categorization, followed by the application of fundamental machine learning and recommendation algorithms. These encompassed content-based, matrix factorization, collaborative user-to-user methodologies, and further integration of clustering techniques. The primary objectives of this comprehensive approach revolve around optimizing job seekers' and recruiters' efficiency while striving to enhance overall accuracy.

Subsequent to the dataset collection elucidated in this paper, our focus transitioned towards the pivotal preprocessing stages. In this phase, an array of Natural Language Processing (NLP) techniques was deployed to extract valuable insights from the job postings. These encompassed essential tasks like text refinement through cleaning, tokenization, and stemming, aiming to refine the dataset for subsequent analyses. Furthermore, we incorporated advanced techniques, including sentiment analysis, aiming to capture nuanced information embedded within the job descriptions.

* 1. *PRE-PROCESSING*

The initial step within the preprocessing approach entails the application of basic statistical methods, encompassing fundamental techniques for data cleaning. An example depicting these statistical methods is presented in Algorithm 1

**Input: DATASET (Raw)**

**Output: DATASET (Processed)**

***Pre-Processing the Data****.*

1. Replace Original Job Location Address With Keywords.
2. Split Experiences Column Into Minimum And Maximum.
3. Separate Education From List Values
4. Separate Industry From List Values
5. Separate Skills From List Values
6. Remove Null Values

**Algorithm 1: Pre-Processing**

*3.1.1 Data Loading*

The code provided commences by importing crucial Python libraries, such as NumPy, Pandas, Matplotlib, and Seaborn for data visualization purposes. Subsequently, it proceeds to load the dataset stored in the "Job\_data.csv" file into a Pandas DataFrame, denoted as "data."

*3.1.2 Data Exploration*

The code snippet presented initiates the exploration of a comprehensive job listings Data Frame, providing an initial overview of its structure and essential characteristics. Utilizing the data.head() function, the code displays the initial rows of the DataFrame, offering a glimpse into the data's composition. Subsequently, data.shape is used to reveal the dimensions of the DataFrame, unveiling a dataset consisting of 22,000 rows and 14 columns. Within these columns, the "jobid" stands as a unique identifier for each job listing, showcasing a wide range of job IDs spanning from approximately 10 billion to 311 billion. This broad numerical range signifies the dataset's diversity and extensive nature.

Furthermore, the "number of positions" column elucidates the availability of positions for each job listing, ranging from 1 to 2000. The variability in the number of positions underscores the heterogeneous nature of job opportunities within the dataset. Calculating the mean, which approximates to 14.56, provides an average insight into the number of positions available across the listings.The calculated value provides an indication of the general distribution of available positions. Transitioning to the experience-related columns, the "Min Experience" column indicates the minimum required experience in years for a given job. This column ranges from 0 years, denoting entry-level positions, to a maximum of 20 years, showcasing the diverse experience prerequisites within the dataset. Similarly, the "Max Experience" column signifies the maximum allowed experience in years for a job, spanning from 0 to 30 years. The mean maximum experience, which is approximately 7.89 years, serves as a measure of central tendency, offering insights into the typical upper bounds of experience sought by employers.

*3.1.3 Data Pre-processing*

The code segment divides the "joblocation\_address" column into various address components by employing .str.split(',') and retains this outcome in a new column labeled "joblocation\_address." Subsequently, it expands the DataFrame by exploding the "joblocation\_address" column. This procedure entails disassembling the lists of addresses into distinct rows, broadening the DataFrame's scope to accommodate multiple addresses within each row. These steps, as outlined in Algorithm 1.

* 1. *Data Analysis*

Subsequent to the expansion of the "joblocation\_address" column, the code proceeds to present the top 20 frequently occurring address components alongside their respective counts. This is accomplished by utilizing data['joblocation\_adress'].value\_counts()[:20]. This particular step offers an extensive understanding of the distribution of address components within the dataset, providing a deeper insight into the most prevalent job locations depicted in the data.

By showcasing the top 20 most frequent address components and their corresponding frequencies, the analysis aims to uncover geographical patterns and concentrations evident in the job listings. This data can play a crucial role in recognizing regions with heightened job availability or specific areas that attract a higher number of employment opportunities. Consequently, this aids in making strategic decisions for both job seekers and recruiters.

* + 1. *Statistical Description of Output*

The original DataFrame, consisting of 22,000 rows and 14 columns, underwent a transformative phase by expanding the "joblocation\_address" column. This expansion resulted in an increased number of rows, facilitating a more comprehensive exploration of job listings based on various address components. The code's execution effectively presents the top 20 address components and their corresponding frequency counts, offering valuable insights into the geographical distribution depicted in the job listings.

However, the showcased outcomes highlight discrepancies in address components, revealing variations such as different spellings or formats for the same location (e.g., "Bengaluru" and "Bengaluru/Bangalore," "Mumbai" and "Mumbai "). This observation indicates potential data inconsistencies that may necessitate cleaning or consolidation efforts to ensure uniformity across the dataset. The statistical summary presented in the code serves as an initial overview of the fundamental operations and data exploration performed, providing a glimpse into the distribution of address components and laying the groundwork for subsequent analysis.

Based on the specific objectives of the analysis, further steps for data cleaning and consolidation may be imperative to rectify discrepancies and bolster the accuracy of subsequent analyses. This preliminary exploration underscores the critical role of meticulous data preprocessing in ensuring the reliability and consistency of the dataset. It sets the foundation for more robust insights into the job listings and their geographical attributes, emphasizing the importance of a thorough data preparation process for meaningful and reliable results.

* 1. *Regression Analysis*

The model showcases a near-perfect fit with an R-squared value of 1.000, indicating potential overfitting. The elevated adjusted R-squared value further suggests a strong fit but also raises concerns about overfitting. The overall model demonstrates high significance, evident from the remarkably high F-statistic. However, individual variables exhibit varying levels of significance, with certain variables (such as x3, x16, x18) displaying relatively low p-values, potentially lacking statistical significance. Among the coefficients, "const" and "x2" demonstrate notably strong positive relationships.

The residual autocorrelation is relatively low (0.273). However, indications of non-normality in the residuals emerge, highlighted by significant p-values from the Jarque-Bera and Omnibus tests. Therefore, it is advisable to conduct further model refinement and diagnostics to address potential issues.

This analysis underscores the need for caution in interpreting the model's performance. Despite the high overall significance and strong fit, concerns such as potential overfitting and variable significance disparities necessitate a more in-depth investigation and refinement of the model..

The detection of non-normality in residuals, as indicated by significant p-values in diagnostic tests, highlights the necessity for thorough model refinement to ensure its reliability and predictive accuracy. Addressing overfitting and variable significance disparities through refined diagnostics remains crucial to enhance the model's robustness and reliability in real-world predictions.

**INPUT: PRE-PROCSSED DATA**

**OUTPUT: REGRESSION TABLE**

***PROCEDURE***

1. Import necessary libraries, including LabelEncoder from scikit-learn.
2. Apply LabelEncoder to transform string values in the DataFrame "df" into integer values, and store the result in a new DataFrame "st."
3. Assign the transformed DataFrame "st" back to "df."
4. Extract the features (independent variables) into "X" and the target variable (dependent variable) into "y" from the DataFrame "df."
5. Split the data into training and testing sets using the train\_test\_split function from scikit-learn.
6. Initialize a LinearRegression model as "regressor" from scikit-learn.
7. Fit the LinearRegression model to the training data using "X\_train" and "y\_train."
8. Predict the target variable using the test data and store the predictions in "y\_pred."
9. Calculate the mean squared error (MSE) and the R-squared (coefficient of determination) to evaluate the model's performance.
10. Import additional necessary libraries, including statsmodels and statsmodels.api.
11. Create a new variable "regressorlm" and set it as equal to the "X\_train" dataset.
12. Add a constant term to "regressorlm" to account for the intercept in the linear regression model.
13. Fit an Ordinary Least Squares (OLS) model to the training data using "y\_train" and "regressorlm."
14. Print a summary of the regression results, which includes statistical information about the model parameters and their significance

**Algorithm 2: Linear Regression**

Standard Errors assume that the covariance matrix of the errors is correctly specified. The smallest eigenvalue is 1.12e-28. This might indicate that there are strong multi collinearity problems or that the design matrix is singular.

IV. VISUALS & ANALYTICS

Visualisations are critical in uncovering complicated patterns, trends, and correlations hidden inside data by providing a clear and meaningful depiction of complex information. Data visualisations promote intuitive understanding by using graphical representations and charts, assisting decision-making processes, and revealing interesting insights. Their use in evaluating algorithmic outputs or statistical summaries can improve model comprehension and emphasise nuances, leading rigorous refinement and informed decision-making in a variety of disciplines. Following the execution of Algorithm-2 on the pre-processed dataset, a thorough regression summary table is produced, displaying a clear depiction of the model's performance as well as the correlations between independent and dependent variables. Despite the model's apparent potential as a result of its apparent perfect fit and overall relevance, its incorporation into a career advice recommendation system demands painstaking modification and study. Personalization, adaptation, and addressing the complexities of individual career trajectories remain critical for providing meaningful and useful advice to consumers seeking professional guidance.

* 1. *Analysis on job, experience*

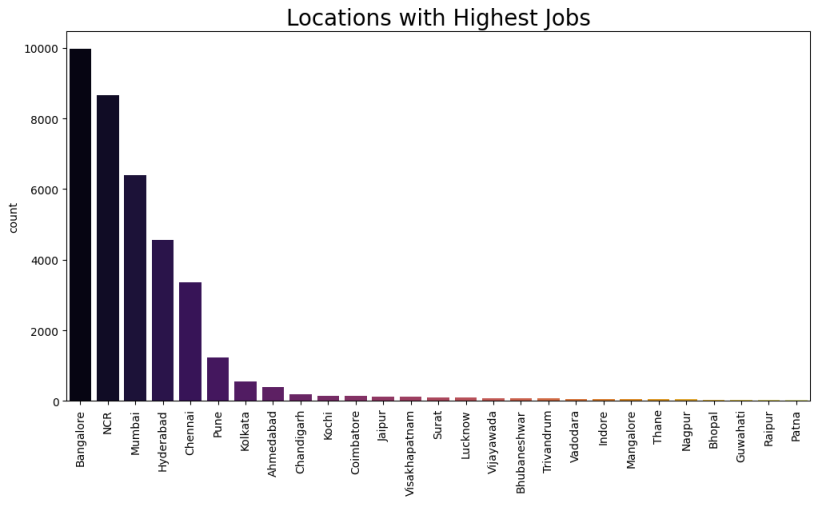
Text analytics empowers the recommendation system to personalize suggestions based on a user's specific skills, qualifications, and objectives. By meticulously considering nuanced details extracted from the text, the system can craft tailored recommendations aligning with each user's distinct profile.

*4.1.1 ANALYTICS ON JOB*

An extensive examination of job distribution across India reveals a concentrated landscape where the top 10 locations account for over 60% of available jobs. A distinct regional divide becomes evident, notably between the northern and western regions, attributed to historical development, infrastructure, and resource accessibility, which have historically influenced economic activities and job prospects.

Noteworthy, shifts in economic dynamics are observable in smaller cities like Chandigarh, Kochi, and Coimbatore, signalling a notable upsurge in job opportunities. This trend suggests a gradual decentralization of economic development, indicating a positive trajectory in diversifying job markets nationwide. The graph serves as a pivotal snapshot, spotlighting key economic hubs and underscoring existing discrepancies in job distribution. These insights offer crucial guidance for policymakers, facilitating strategies aimed at fostering inclusive economic growth and job creation across all regions of the country.

Addressing these imbalances is vital for policymakers to cultivate a more balanced job landscape, aligning with broader national development objectives. The insights gleaned from Figure-1 serve as a vital tool, empowering policymakers to devise interventions promoting equitable economic growth and mitigating regional disparities in employment opportunities.

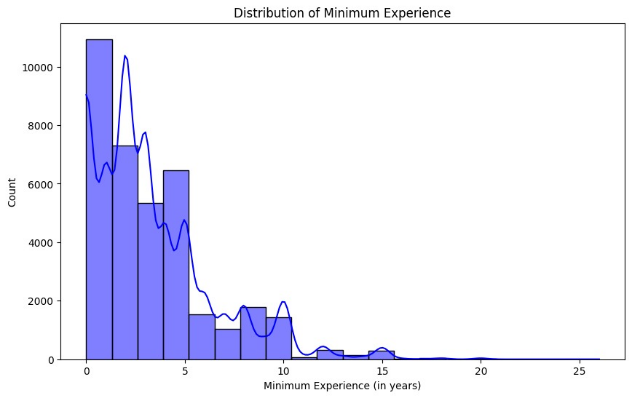
**Figure 1: Highest jobs vacant Location in dataset**

*4.1.2 ANALYTICS ON EXPERIENCES*

In the realm of yeast-related professions, a comprehensive analysis reveals a notable transformation and maturation of the job market. The depicted graph illustrates a discernible shift away from entry-level positions, signalling an increasing demand for seasoned professionals in yeast-related fields. The insights gleaned from an in-depth analysis of professional journeys and expertise within a specific field or industry. Analytical approach involves scrutinizing trends, shifts, and the evolving demand for varying levels of experience trend accentuates the growing recognition of specialized skills and expertise, highlighting the necessity for candidates with substantial experience.

An intriguing aspect unveiled by the data is the expanding reach of yeast professionals beyond the conventional food and beverage sector. Their expertise is now sought after across diverse industries such as pharmaceuticals, biofuels, and cosmetics. This broadening application indicates the growing relevance of yeast-related skills in various scientific and commercial domains.

Given the competitive nature of the yeast job market, it is expected that this trend will persist, driven by on-going expansion and specialization within the field. Consequently, individuals aspiring to excel in yeast-related industries are advised to focus on cultivating specific skills and acquiring relevant experience valued by employers. This may involve pursuing specialized academic degrees, engaging in internships, or participating in relevant volunteer opportunities aligned with the evolving demands of the yeast job market.

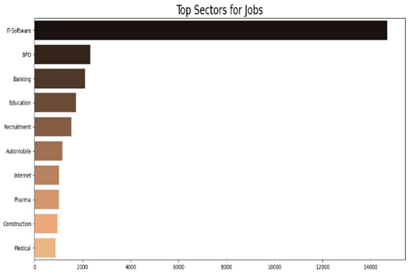


**Figure 2: Explain the Experience Distribution**

In summary, Figure-2 provides an insightful depiction of the evolving landscape within yeast-related professions. It underscores the escalating demand for experienced professionals and the widening spectrum of industries seeking their expertise. The competitive nature of this job market underscores the necessity for strategic career development; ensuring individuals remain well-equipped to meet the evolving demands of employers in this vibrant and expanding industry.

*4.1.3 ANALYTICS ON SECTOR*

Within Figure-3, a compelling storyline emerges, urging the implementation of strategic actions aligned with the evolving job market's dynamics to ensure continued economic growth. The highlighted priority emphasizes investing in customized education and training programs. These programs aim to equip the workforce with essential skills tailored for upcoming high-skill sectors. This strategic approach not only meets immediate industry needs but also ensures the long-term sustainability and competitiveness of the workforce. Moreover, the graph accentuates the promotion of innovation and entrepreneurship within key job sectors. Encouraging an innovative culture can act as a driving force for economic growth while creating fresh avenues for employment opportunities. This emphasis on nurturing dynamism and an entrepreneurial mind-set aligns with the ever-changing job market, facilitating adaptability and progress amid evolving demands. An essential facet highlighted in the data is the importance of cultivating an attractive environment to draw and retain talent, including international workers. This involves strategic investments in infrastructure and enhancements in quality of life, critical factors influencing talent retention and the overall vibrancy of the job market.

**Figure 3: Top placed Sector jobs available**

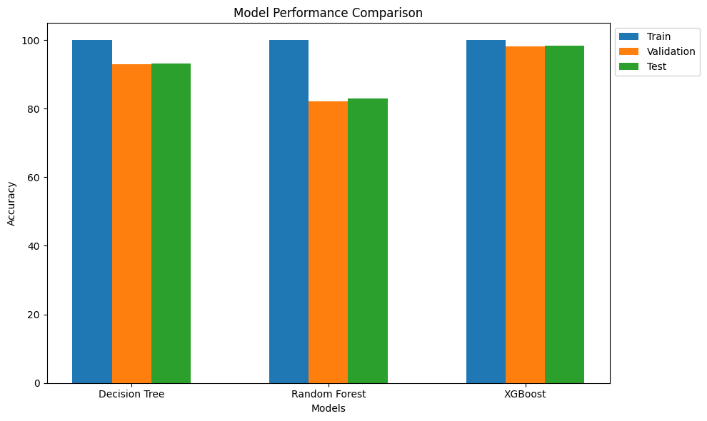
**4.2 *OVERALL, SKILL ANALYTICS***

The rising demand in the job market signals an increasing focus on comprehensive skill sets, as depicted in the graph. The amplified need for improved overall skills stems from various factors, including heightened job intricacy, global economic integration, and swift technological advancements.

The evolution of the job market necessitates workers to prioritize enhancing their overall skills to remain competitive, particularly in the face of technological advancements. The ability to continuously up skill and adapt is paramount in meeting the shifting demands of this dynamic job market. Strategic investments in education and training programs aimed at nurturing comprehensive skill sets are crucial, not only for individual career progression but also for fostering overall economic growth.

V. MODELS

5.1 *ALGORITHMS*

In the domain of job guidance, an array of diverse machine learning classifiers are available to facilitate well-informed decision-making and personalized recommendations for individuals navigating the intricate landscape of professional choices. The Decision Tree Classifier ('dt') operates via an attribute-based data splitting process, offering transparency into the decision-making logic and aiding in matching individuals with suitable career paths. The Random Forest Classifier ('rf') amalgamates insights from multiple decision trees to enhance prediction accuracy, providing robust career recommendations by considering various factors. Meanwhile, the XGBoost Classifier ('xgb'), known for its efficiency and predictive prowess, excels in identifying optimal career paths by evaluating multiple attributes and aligning them with suitable job opportunities.

**Fig 4: Machine learning algorithm comparison**

Logistic Regression ('logistic') is a statistical method particularly valuable in this context, predicting binary outcomes to assess the likelihood of an individual's success in specific jobs or industries. Gradient Boosting Classifier ('gb') combines weak models to provide refined career suggestions, considering a broad spectrum of factors for identifying the best career options. AdaBoost Classifier ('adaboost') emphasizes relevant attributes and patterns to offer effective career path recommendations. The Support Vector Classifier ('svc'), focusing on finding optimal decision boundaries, plays a crucial role in distinguishing among career choices, especially in high-dimensional spaces.

|  |  |
| --- | --- |
| **Parameters** | **Value in** |
| Accuracy | 98% |
| Precision | 97% |
| Recall | 91% |

**Table 1: Explains best result ML algorithm XGB**

Together, these classifiers form a comprehensive toolkit, leveraging machine learning to offer tailored career guidance by considering individual skills, qualifications, and preferences for more informed decision-making in pursuit of professional success. This analysis reveals the potential of different classifiers in job guidance but highlights the need for further evaluation and consideration of factors such as training time, data size, and precision to tailor the recommendation system effectively.

The evaluation of various machine learning classifiers in Figure-4 sheds light on their potential for career guidance in recommendation systems. Among the models assessed, the Decision Tree Classifier showcases exceptional performance, boasting a perfect F1 score and precision, signifying its capability to offer highly accurate career recommendations. The Random Forest Classifier, although slightly longer to train, also provides reasonably accurate guidance with a respectable F1 score and precision. On the other hand, while the XGBoost Classifier demonstrates high accuracy at 98.214%, it shows a slightly lower F1 score.

However, the Logistic Regression, Gradient Boosting Classifier, AdaBoost Classifier, and Support Vector Classifier lack available training and performance metrics, necessitating more information for a comprehensive assessment of their potential. Ultimately, the choice of classifier should be tailored based on factors like training time, dataset size, and desired precision to create a recommendation system meeting the specific needs of users seeking job guidance.

**INPUT: PRE-PROCESSED DATA**

**MODEL: MACHINE LEARNING MODEL (XGB)**

**OUTPUT: JOB TITLE RECOMMENDATION ‘S**

***PROCEDURE***

# They need to be recommended on priority basis ==> post date

recommendation\_df = data[data['industry'] == pred].sort\_values(by='postdate', ascending=True)

n\_recommendations = 100

# Show the first 100 recommendations

recommendations = recommendation\_df.iloc[:,:n\_recommendations]

recommendations  
**Algorithm 2: Machine learning XGB**

*5.2 RECOMMENDATION ALGORITHMS*

Several advanced methods are utilized to elevate user engagement and deliver personalized recommendations. Collaborative Filtering, a widely adopted technique, encompasses both User-Based Collaborative Filtering and Item-Based Collaborative Filtering. The former suggests items by identifying users with akin preferences, leveraging their actions and preferences to propose suggestions. Meanwhile, the latter assesses the similarity between items a user has interacted with and other items in the system, providing recommendations based on observed user engagement patterns.

*5.2.1 Matrix Factorization*

Matrix Factorization involves breaking down the user-item interaction matrix into lower-dimensional matrices, unveiling latent factors that capture user preferences and item patterns. This understanding enables precise recommendations by discerning subtle user-item relationships

*5.2.2 Content-Based Filtering*

Contrasting collaborative approaches, Content-Based Filtering recommends items based on their content and user profiles. By aligning item features with user preferences, it offers tailored suggestions that resonate with individual tastes.

*5.2.3 Context-Aware Recommendation*

Context-Aware Recommendation goes beyond typical methods by incorporating contextual cues like location, time, or device used. This approach fine-tunes suggestions to cater to the user's immediate situation and requirements.

*5.2.4 Collaborative Filtering Algorithm*

Collaborative Filtering Algorithm, a pivotal method in recommendation systems, facilitates the generation of personalized suggestions by identifying patterns in user behaviour. This approach primarily relies on analysing user interactions with items, such as clicks, views, or applications, to establish connections between users with similar preferences. By leveraging these connections, the algorithm suggests items that align with a user's preferences but have not yet been explored. This personalized approach enhances user engagement and satisfaction by offering tailored recommendations tailored to individual tastes and preferences

1. *Creation of User-Job Interaction Matrix* The foundational step involves constructing a matrix reflecting user interactions with jobs. Implicit feedback (clicks, views, applications) marks interactions as 1 in the matrix, forming a comprehensive representation.
2. *Computation of User Similarity* Cosine similarity metrics are computed to measure user resemblance based on interaction patterns. Users with higher similarity scores share analogous job preferences, forming the basis for personalized recommendations.
3. *Personal User-Centric Recommendation system* The 'user\_collaborative\_filtering'function identifies users with akin preferences, excluding the target user. By recommending unexplored jobs that similar users have engaged with, it offers personalized and contextually relevant suggestions.

The recommendation process is meticulously orchestrated, as the algorithm identifies the top users with similar preferences, excluding the target user. For each of these similar users, it identifies jobs they've interacted with, creating a tailored list of potential preferences for the target user. These unexplored job opportunities are then presented as personalized recommendations, ensuring relevance and contextually for the user.

*5.2.1 Practical Application*

Collaborative filtering methods like this one are extensively utilized in various recommendation systems, spanning job suggestions, movie selections, and product endorsements. They effectively furnish users with personalized recommendations by analysing behaviours and preferences akin to those of other users. In the realm of job recommendations, this method aids job seekers in exploring roles aligned with their interests, expertise, and competencies.

**INPUT: DATASET OF USER PRISONIZATION’S  
OUTPUT: RECOMMEND TITLE’S**

***PROCEDURE***

# Pseudocode for User-Based Collaborative Filtering

# Step 1: Create a user-job interaction matrix

for each user in data:

for each job in data:

if user interacted with job:

user\_job\_matrix[user][job] = 1

else:

user\_job\_matrix[user][job] = 0

# Step 2: Calculate user similarity using cosine similarity

user\_similarity = cosine\_similarity(user\_job\_matrix)

# Step 3: Function to recommend jobs to a user based on user similarity

function user\_collaborative\_filtering(user\_index, top\_n):

user\_sim\_scores = []

for each user in user\_similarity:

similarity\_score = calculate\_cosine\_similarity(user\_index, user)

user\_sim\_scores.append((user, similarity\_score))

user\_sim\_scores = sort(user\_sim\_scores, key=lambda x: x[1], reverse=True)

top\_sim\_users = user\_sim\_scores[1:top\_n + 1] # Exclude the user itself

recommended\_jobs = set()

for each similar\_user in top\_sim\_users:

similar\_user\_jobs = get\_jobs\_interacted\_by\_user(similar\_user)

for each job in similar\_user\_jobs:

if user\_job\_matrix[user\_index][job] == 0:

# Job not interacted by the target user

recommended\_jobs.add(job)

print("User Index:", user\_index)

print("Top Similar Users:", top\_sim\_users)

print("Recommended Job Indices:", recommended\_jobs)

return first top\_n elements of recommended\_jobs

# Example: Recommend jobs for a specific user

user\_index = 10

recommended\_jobs = user\_collaborative\_filtering(user\_index, top\_n=5)

print("Recommended Jobs:")

*OUTPUT:*

User Index: 10

Top Similar Users: [(1, 1.0000000000000007), (2, 1.0000000000000007), (3, 1.0000000000000007), (4, 1.0000000000000007), (5, 1.0000000000000007)]

**Algorithm 3: Collaborative Filtering**

In collaborative filtering, the recommendations hinge upon interactions among users who exhibit similar behaviour patterns. However, if there are no comparable users who have engaged with jobs the target user hasn't explored, it might lead to an absence of suggestions. To refine the effectiveness: augment the dataset by increasing interactions, explore diverse similarity metrics like cosine similarity or Pearson correlation, consider implicit feedback like views or clicks for better inferences, and adjust the number of recommended jobs (n parameter) to ensure a more extensive set of suggestions. Collaborative filtering might not prove significantly more effective compared to content-based filtering when applying these strategies for enhanced job recommendations

***5.2.2 CONTENT BASED RECOMMENDATION***

Content-based recommendation involves a personalized approach that relies on item attributes and user preferences to propose relevant suggestions. In this method, recommendations are formulated by considering the characteristics of the items and the past preferences of users.

**INPUT: DATASET OF USER PRISONIZATION’S**

**OUTPUT: RECOMMEND TITLE’S**

*PROCEDURE*

Feature Extraction:

Initialize a TF-IDF vectorizer.

Transform job descriptions and skills into TF-IDF feature vectors.

Extract experience features (e.g., lower and upper bounds).

Batch Processing:

Define a batch size for memory efficiency.

Initialize variables and arrays for storing feature vectors.

Divide the data into batches and process each batch.

Combine TF-IDF and experience features.

Similarity Calculation:

Calculate cosine similarities between combined feature vectors.

cosine\_similarities = cosine\_similarity(combined\_features, combined\_features)

*User Profile Similarity Calculation:* Define a function to calculate similarity between a user profile and job listings.

Prepare the user's profile feature vector by combining TF-IDF and experience features.

Calculate cosine similarity between the user profile and all job listings.

*Recommendations:* Set the number of recommendations (e.g., 5).

Select a random user profile from the dataset.

Calculate the similarity between the user profile and all job listings.

Rank the job listings by similarity scores.

Print the recommendations.

n\_recommendations = min(5, len(data) - 1)

random\_user\_profile = data.sample(n=1).iloc[0]

similarity\_scores= calculate\_similarity(random\_user\_profile, combined\_features)

recommended\_indices = similarity\_scores.argsort(axis=0)[-n\_recommendations:]

print("User Profile:")

print(f"Job Title: {random\_user\_profile['jobtitle']}")

print(f"JobDescription: {random\_user\_profile['jobdescription']}\n")

print("Recommended Jobs:")

for idx in recommended\_indices:

job\_title = data.iloc[idx]['jobtitle']

print(f"Job Title: {job\_title}")

print(f"Similarity\_Score: {similarity\_scores[idx][0]}\n")

**OUTPUT**

User Profile:

Job Title: Production Executive (garments Preferably)

Name: jobtitle, dtype: object

Job Title: 2483 Senior Manager, IT Applications

Name: jobtitle, dtype: object

Similarity Score: [0.96278719]

Job Title: 21147 AVP Financial Planning and Analysis - It/ites/...

Name: jobtitle, dtype: object

Similarity Score: [0.96287655]

Job Title: 9673 Opening for Senior Director in Bangalore

Name: jobtitle, dtype: object

Similarity Score: [0.96324824]

Job Title: 14457 GM Merchandising (Woven Garment) FOR A Factory...

Name: jobtitle, dtype: object

Similarity Score: [0.96402535]

Job Title: 10246 Production Executive (garments Preferably)

**Algorithm 4: Content Based Filtering**

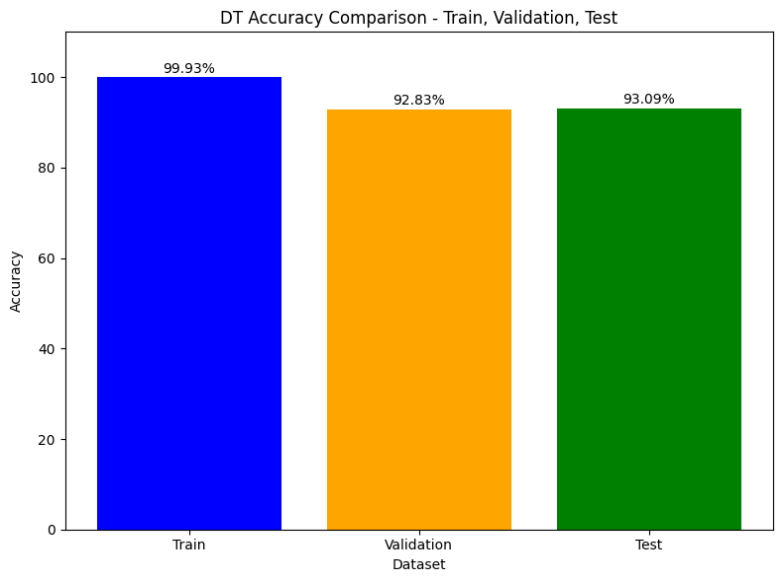
Job Description: This job involves coordinating with the sourcing department to ensure timely receipt of quality materials for production. Responsibilities include coordinating the first bulk production inspection, conducting technical merchandising team meetings, managing factory shipments, and maintaining audit files. The role requires coordinating various factory reports, facilitating meetings, and overseeing production processes. The salary ranges from INR 225,000 to 275,000 per annum in the textile and garment industry within the production and manufacturing sector. The desired candidate should possess good interpersonal, coordination, and reporting skills.

Company Profile: Gokaldas Image Pvt Ltd is one of the largest integrated clothing corporations, highly ranked in the marketing, retailing, and exporting sectors within the clothing industry.

VI. Results and Experiments

The outcomes of the conducted experiments light on the performance of various models utilized for job recommendation tasks. The Decision Tree (dt) model demonstrates exceptional performance on the training set, boasting a near-perfect accuracy rate of 99.934%. However, upon evaluation on the validation and test sets, its accuracy diminishes, recording scores of 92.831% and 93.094%, respectively. Delving deeper into the classification reports reveals that specific classes exhibit lower precision, recall, and F1-score values, implying potential challenges the model encounters with certain classes. This pattern suggests a likelihood of over fitting to the training data, signalling potential limitations in generalizing to new, unseen data.

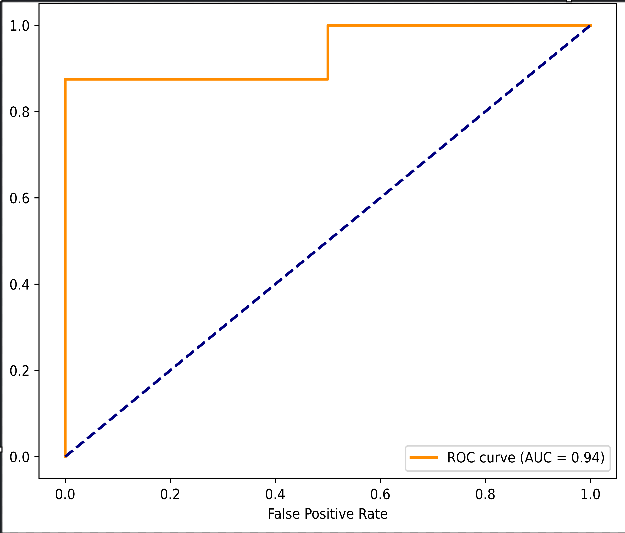
Similarly, the Random Forest (rf) model mirrors this trend. While displaying impressive accuracy during training (99.934%), its performance declines when assessed on the validation and test sets, recording accuracies of 82.077% and 82.973%, respectively. Once again, the classification reports uncover difficulties with specific classes, reinforcing the notion of potential issues with the model's performance across various job categories.



**Figure 5: Classification report of Decision Tree**

The Decision Tree and Random Forest models demonstrate potential signs of over fitting to the training data, evident in the discrepancies between their performance on training versus validation/test datasets. To mitigate over fitting, a strategic adjustment of hyper parameters—such as the decision tree's maximum depth [Figure 5] or the number of trees in the Random Forest—is advisable. Furthermore, delving into feature importance analysis for both models can offer valuable insights into significant predictors, guiding efforts in feature selection or engineering to enhance interpretability and potentially bolster overall performance. Additionally, a thorough examination of class imbalances within the dataset holds potential in refining predictive accuracy, particularly for minority classes.

Evaluating the model's performance metrics—precision, recall, and F1 score—yields insightful results. A precision score of 0.6667 signifies that roughly 66.67% of instances identified as positive align with true positives. Simultaneously, a recall of 0.6667 indicates the model's capability to capture around 66.67% of all actual positive instances. The derived F1 score of 0.6667 represents a balanced trade-off between precision and recall. To elevate model effectiveness, an in-depth analysis of misclassifications is recommended to discern underlying patterns or specific scenarios posing challenges for the model. Potential enhancements may involve refining hyper parameters, adjusting decision thresholds, or incorporating additional relevant features. Addressing potential class imbalance, if identified, holds promise in bolstering overall model efficacy. Consistent model evaluation and refinement guided by real-world feedback are pivotal for ensuring sustained effectiveness in practical applications.



**Figure 6: Receiver Operating Characteristic (ROC) vs. Cosine similarity Curve Report**

The analysis of the ROC curve [Figure 6] suggests that the recommendation system is performing commendably, exhibiting an AUC of 0.94. This score indicates the system's proficiency in distinguishing between relevant and irrelevant items with a notably high degree of accuracy. While a perfect recommendation system achieves an AUC of 1.0, signifying consistent ranking of relevant items above irrelevant ones, a random system would score 0.5, indicating random guesses in item relevance. Hence, an AUC of 0.94 implies that the recommendation system performs exceptionally well, effectively identifying and suggesting relevant items to users.

The ROC curve illustrates the system's impressive performance, affirming its ability to accurately recommend relevant items with high precision. To enhance its practical application, further model refinement strategies such as hyper parameter tuning and addressing class imbalances should be considered for future enhancements in its performance.

The high AUC score of 0.94 reflects the recommendation system's robustness in identifying relevant items. However, delving deeper into model refinement could fortify its performance. Fine-tuning hyper parameters, such as adjusting thresholds or optimizing model settings may optimize the system's precision and recall metrics. Additionally, addressing class imbalances, if present in the dataset, could be crucial in improving the system's accuracy, especially for underrepresented categories. Moreover, exploring ensemble techniques or employing hybrid recommendation approaches, combining collaborative filtering with content-based or hybrid models, might contribute to a more comprehensive and accurate recommendation system. Integrating feedback mechanisms for continuous learning and adapting to evolving user preferences would further augment the system's efficacy and relevance.

Regular monitoring and evaluation of the recommendation system in real-world scenarios are vital for identifying areas of improvement and ensuring sustained high performance. By continually refining the model and adapting it to changing user behaviour’s and preferences, the recommendation system can consistently deliver precise and tailored recommendations, ultimately enhancing user satisfaction and engagement.

VII CONCLUSION

We did a thorough examination throughout our research and evaluation of recommendation algorithms for improving employment guidance to discover the most effective way for supporting persons in their career aspirations. Among the algorithms evaluated, Content-Based Filtering performed the best in this study. Its extraordinary success was due to its capacity to thoroughly assess job descriptions, abilities, and experience levels, resulting in highly personalised job recommendations fit with users' qualifications and objectives. Content-Based The strength of filtering is its granularity of analysis, which allows for personalised matching between job seekers and positions that best match their capabilities and career goals. This technique improves the relevance and quality of career advice by personalising recommendations to particular users rather than presenting generic suggestions.

Our research journey demonstrated the usefulness of Content-Based Filtering in offering personalised and data-driven career development assistance. This strategy has the potential to dramatically improve users' job search experiences by focusing on individual qualities and doing detailed assessments of job descriptions, allowing people to make well-informed career choices fit with their specific abilities and goals. Continuing to explore and enhance these algorithms in combination with user feedback is critical for improving the efficacy of solutions targeted at improving job suggestions.

VIII FUTURE WORK

The proposed development of the recommendation system includes a wide range of functions designed to improve user involvement and encourage further improvements. First and foremost, the integration of User Feedback is critical, allowing users to give thoughts on job recommendations, skill gap evaluations, and interview inquiries. This iterative feedback method supports continuous system development, ensuring that recommendations improve over time to better correspond with user preferences and requirements. Another critical feature is Industry Trends Analysis, which provides real-time insights into industry movements and employment market dynamics. This proactive strategy portrays the system as a dynamic resource that is responsive to changing job market demands.

Job-Specific Questions add value by preparing users for job applications with specialised interview or assessment questions for each proposed position. This not only helps with practise, but it also increases user confidence when navigating the job application process. Furthermore, Skill Verification allows users to enter their abilities, qualifications, and certifications, which the system then validates and highlights. This function boosts the legitimacy of users' profiles, potentially enhancing their job application success. The system's utility is expanded by professional Pathway Suggestions, which go beyond simple employment choices to present users with a vision for long-term professional advancement. The system serves as a holistic career advising tool, providing potential pathways and growth chances by aligning recommendations with users' interests and abilities.

Skill Gap Analysis increases the system's usability by assessing users' skill sets for each recommended job and indicating areas that need emphasis. This feature provides users with a clear understanding of the abilities required for specific jobs, allowing them to make educated decisions about skill development and training. As these innovative components are incorporated into the system's architecture, it not only becomes a valuable resource for job seekers and career changers, but it also positions itself as an adaptive, user-centric platform actively contributing to users' professional development and success in the ever-changing job landscape.

VII REFERENCES

[1] Vignesh S, Shivani Priyanka C, Shree Manju H, Mythili K.

"An Intelligent Career Guidance System Using Machine

Learning." 2021**.**

[2] Manar Qamhieh, Haya Sammaneh, Mona Nabil

Demaidi. "Pcrs: Personalized Career-Path Recommender

System For Engineering Students." Ieee.

[3] Manar Qamhieh, Haya Sammaneh, Mona Nabil

Demaidi. "Pcrs: Personalized Career-Path Recommender

System For Engineering Students." Ieee.

[4] Tanya V Yadalam, Vaishnavi M Gowda, Vanditha Shiva

Kumar, Disha Girish. "Career Recommendation Systems

Using Content- Based Filtering." 2020.

[5] Wenxing Hong, Siting Zheng, Huan Wang. "Dynamic User

Profile-Based Job Recommender System." 2013.

[6] Anitha Anandhan, Liyana Shuib, Maizatul

Akmar Ismail, Ghulam Mujtaba. "Social Media

Recommender Systems: Review And Open Research Issues."

2017.

[7] Anika Gupta, Dr. Deepak Garg. "Applying Data Mining

Techniques In Job Recommender System For Considering

Candidate Job Preferences." 2014.

[8] Yingya Zhang, Cheng Yang, Zhixiang Niu. "A Research Of Job

Recommendation System Based On Collaborative Filtering."

2014.

[9] Luis M. De Campos, Juan M. Fernández-Luna, Juan F. Huete,

Miguel A. Rueda-Morales. "Combining Content-Based And

Collaborative Recommendations: A Hybrid Approach Based

On Bayesian Networks." 2010.

[10] Zeshan Fayyaz, Mahsa Ebrahimian, Dina Nawara, Ahmed

Ibrahim, Rasha Kashef. "Recommendation Systems:

Algorithms,Challenges, Metrics, And Business Opportunities."

2020.

[11] C. H. Chen, S. D. Mayanglambam, F. Y. Hsu, C. Y. Lu, H. M.

Lee, J. M. Ho. "Novelty Recommendation Using Citation

Authority Diffusion." Proceedings - 2011 Conference On

Technologies And Applications Of Artificial Intelligence, Taai   
 2011, 2011, Pp. 126–131.

[12] W. Zhang, T. Yoshida, X. Tang. "A Comparative Study Of

Tf\*Idf, Lsi And Multi-Words For Text Classification."

Expert Syst. Appl., Vol. 38, No. 3, Pp. 2758–2765, 2011.

[13] J. Ramos. "Using Tf-Idf To Determine Word Relevance In

Document Queries." Proc. First Instr. Conf. Mach. Learn., Pp. 1–

4, 2003.

[14] T. Kenter, M. De Rijke. "Short Text Similarity With Word

Embeddings Categories And Subject Descriptors." Proc. 24th

Acm Int. Conf. Inf. Knowl. Manag. (Cikm 2015), Pp. 1411–

1420, 2015.

[15] S. Albitar, S. Fournier, B. Espinasse. "An Effective Tf/Idf-Based

Text-To-Text Semantic Similarity Measure For Text

Classification." Web Inf. Syst. Eng. - Wise 2014, Pt I, Vol. 8786,

Pp. 105–114, 2014.

[16] C.-H. Huang, J. Yin, F. Hou. "A Text Similarity Measurement

Combining Word Semantic Information With Tf-Idf Method."

Chinese Journal Of Computers, 2011.

[17] S. H. Strogatz. "Exploring Complex Networks." Nature, Vol.

410, No. 6825, Pp. 268–276, 2001.

[18] M. E. J. Newman. "The Structure And Function Of Complex

Networks." Siam Review, Vol. 45, No. 2. Pp. 167–256, 2003.

[19] M. Faloutsos, P. Faloutsos, C. Faloutsos. "On Power-Law

Relationships Of The Internet Topology." In Sigcomm, 1999, Pp.

251–262.

[20] R. Albert, A. L. Barabasi. "Statistical Mechanics Of Complex

Networks." Rev. Mod. Phys., Vol. 74, No. 1, Pp. 47–97, 2002.