International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

# Improved Inductive Learning Approach -5 (IILA-5) in Distributed System

**Ravita Mishra[1], Bhushankumar Nemade[2]*, Kamal Shah[3], Pravin Jangid[4]**

**Submitted**: 24/05/2023 **Revised**: 06/07/2023 **Accepted**: 25/07/2023

**Abstract***:* The job recommender system is a proficient data-driven application that utilizes inductive learning to create a comprehensive set of classifying rules that effectively matches suitable jobs and skills to both candidates and recruiters. The system faces difficulties in real-world scenarios because the data it deals with frequently contains noise, insufficiency, activeness, unwanted attributes, continuous variables, and missing values. As a result, generating accurate rules from such inconsistent datasets becomes a difficult task. To address these issues, the system employs the novel Inductive Learning Approach-5 (ILA-5) algorithm. ILA-5 is specifically designed to address these complexities by removing unnecessary and irrelevant rules while improving prediction accuracy on unseen training data. The algorithm employs an iterative approach, actively exploring rules that improve training sample arrangement. When a set of rules is generated, the system labels the corresponding training data samples. Following cycles, the algorithm efficiently rejects rules that do not contribute significantly to the overall accuracy of the recommendations. The ILA-5 algorithm's effectiveness has been rigorously tested on two distinct datasets, namely CareerBuilder and Niti Aayog's dataset. The astounding results show its ability to generate rules with an impressive 91% accuracy for job recommendation. Furthermore, the algorithm has excellent scalability, allowing it to handle large datasets without sacrificing efficiency. Overall, the job recommender system with ILA-5 is a cutting-edge solution that provides powerful, accurate, and scalable job recommendations, benefiting both job seekers and recruiters in the ever-evolving job market.

***Keywords****:* *MVI (Missing Value Imputation), ILA-1 (Inductive Learning Approach-1), ILA-2, ILA-3, ILA-4, ILA-5, FastILA.*

## 1. Introduction

Inductive learning, which focuses on identifying broad descriptions, is a key stage in data mining. In supervised machine learning, inductive learning is provided with a collection of training samples, where each sample is defined by a vector of attribute values and a class label. It generates a set of rules that covers all potential cases. Inductive learning term characteristics may contain the job id, company name, education, job title, technical skill, and class label, indicating a job recommendation or skill suggestion. Inductive learning algorithms' primary purpose is to maximize classification power on previously unknown test data. To compare ILA performance, we used the classification algorithms ID3 (Iterative Dichotomiser 3) and AQ (Algorithm quasi optimal). ID3 is an overturning technique based on decision trees that conducts stepwise splitting and occasionally overfits due to excessive and irrelevant constraints. It can occasionally influence the categorization of an unknown material. Tree pruning is a common solution to this problem, however it does not work with probabilistic data. Uthaursamy (1991) demonstrated a method for dealing with inconclusive results. datasets. ID3 also fails to work on a bigger sample set. Although the windowing strategy addresses sample issues, decision trees are unable of appropriately categorizing all samples [1, 9].

The AQ method is a machine learning approach that learns via symbolic decision rules. It generates a set of characteristics, value criteria, and specializations while eliminating negative instances. The amount of complexity space in AQ searches corresponds to the actual data, and rules do not work flawlessly on training data. Similarly, CN2 employs the same heuristic method as AQ, but without the necessity for a specific instance. AQ and CN2 rule-driven induction approaches that do not need flow diagrams. Salzberg (1994) demonstrated a decision tree induction approach for numerical value characteristics. Aksoy, 1995, developed an alternative RULES rule induction technique that can classify hidden instances. The RULES algorithm creates an increasing number of rules, making vast volumes of data difficult to manage [1]. The main contribution of this paper is as follows:

1.Compares the existing ILA algorithm with the Proposed ILA algorithm.

2.Results are validated on two different datasets (careerBuilder.com and Niti Ayog's Job data.

3.The existing ILA algorithm works on small datasets with low scale applications. But proposed methodology works on centralized as well as distributed datasets.

[1] *Assistant professor, Vivekanand Education Society's Institute of Technology, Mumbai,*
[2]*Assistant professor, Mukesh Patel School of Technology Management & Engineering, NMIMS University, Mumbai,*
[3]*Professor, Thakur College of Engineering and Technology, Mumbai,*
[4]*Assistant Professor, Shree L R Tiwari College of Engineering (SLRTCE), Mumbai.*
[1]*ravita.mishra@ves.ac.in,* [2]*bnemade@gmail.com,*
[3]*kamal.shah@thakureducation.org,* [4]*pravinjangid@gmail.com*

The paper is organized into four main parts: the first part includes the basic introduction of all existing inductive learning techniques and their specified domain. The second part discussed the literature review of this approach and their summary. The third part includes the proposed methodology with application domain and their casual representation of jobs in various datasets. The last part discusses the various results obtained from these experiments and the conclusion listed in last.

### 1.1. Inductive Learning

In terms of practical application, the ILA rule presentation approach is applicable in the context of data exploration. This is due to the fact that it focuses on one rule at a time, building a rule with its ancestral component that includes the descriptive element. The ILA algorithm is divided into three major classes: ILA-1, ILA-2, and ILA-3, with ILA-4 being the final planned iteration. ILA is in charge of creating a set of structural rules for a set of training samples. It works in an iterative fashion, with each iteration focused on establishing a rule that covers a given class with several training instances. Suggestions for improvement. Figure 1 depicts the progression of the ILA lineage from 1998 to 2022.

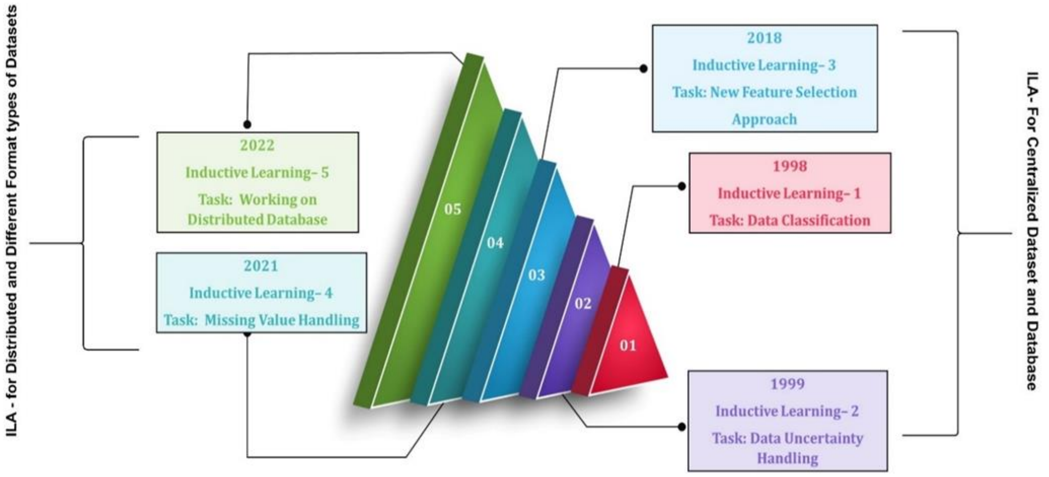

ILA - for Distributed and Different Format types of Datasets
2022
Inductive Learning- 5
Task: Working on Distributed Database
2021
Inductive Learning- 4
Task: Missing Value Handling
05
04
03
02
01
2018
Inductive Learning- 3
Task: New Feature Selection Approach
1998
Inductive Learning- 1
Task: Data Classification
1999
Inductive Learning- 2
Task: Data Uncertainty Handling
ILA- For Centralized Dataset and Database

**Fig 1.** ILA (Inductive Learning Algorithm) Family

The ILA methods project can be used to classify samples into a variety of different categories, such as spam, phishing, malware, and benign. This allows for a high degree of accuracy, which is essential for ensuring that the correct samples are classified.

**1.1.1 ILA-1**: ILA-1 is a supervised and robust method for illustrative data classification, specifically for discrete and symbolic data. The way it works adopts an iterative pattern, with each iteration looking for a rule that covers a large number of training cases. It has the ability to create IF-THEN rules using various algorithms on a sample set. ILA-1 assesses all instances of a given class, identifying the overall qualities while avoiding repetitive and unrelated requirements. The CART algorithm generates rules that are less complicated and more prevalent than the ID3 and AQ methods. In compared to ID3 and AQ, ILA-1 uses a class-centric method to generate more compact rules per class [3]. ILA-1 leverages two factors for rule evaluation. First, it takes into account the number of created rules and the average condition tally. Its goal is to produce a small collection of rules which efficiently classify the samples in the training dataset. Second, it attempts to demonstrate the algorithm's ability to classify unseen data. ILA is a dependable inductive procedure that works with non-contradictory sample sets. Two major issues emerge from the ILA-1 framework: over-fitting and longer learning periods as well.

**1.1.2 ILA-2**: Real-world data has noise, insufficiency, dynamism, unwanted aspects, continuity, and missing information. This clearly underscores the importance of each stage in the process of transforming raw data into valuable insights. When dataset descriptions are insufficient, divergent datasets lack enough substance to infer rules effectively. Inductive learning algorithms are primarily concerned with deducing thorough definitions of a concept from a set of training instances. ILA-2 is a useful tool for data scientists who want to extract accurate and simply understandable rules from datasets. It is a more powerful and efficient rule induction algorithm than ILA-1 and it includes two new evaluation measures to manage the uncertainty and bias in the data. The accuracy of ILA-2 is better than ILA-1, and it classifies unlabelled data fastly, and the learning rate and size of classifier size also improved. The key rule evaluation parameter in ILA-2 is class, dimension and precision. The class dimensions stand for the total number of conditions included in the cluster's rules. The Penalty Factor measures the negative impact of instances mistakenly classified as negative for the structure of the description. FastILA improves the performance of ILA-2 by using a single description to create a new rule, significantly decreasing the processing time from 18 seconds to 11 seconds [6].

**1.1.3 ILA-3**: When dealing with large and noisy datasets, ILA-1 exhibits inefficiency, which is characterized by several unique factors: 1) the number of rule generations, 2) the simplicity of the rules, 3) the usefulness of the created rules in properly detecting fresh samples, and 4) the pace at which rules are generated (algorithm efficiency). To solve this issue, ILA-3 was created and adapted, adding a new feature selection algorithm designed specifically for such cases. ILA-3 takes the method of segmenting instances into secondary tables and constructing sub-tables for particular class values. Each sub-table applies the new CombExclude algorithm, which employs numerous combinations and excludes unnecessary information, storing these exclusions in the ExcludedCombinationsList. This update significantly improves efficiency over the original ILA since it processes the full dataset without removing any combinations [10]. This method provides a more compact representation of the target notion, improving future classification accuracy. It removes unnecessary, redundant, or noisy data, providing immediate benefits to applications such as improving knowledge base algorithms and extraction performance, including predicting accuracy and consistency. The collected data reveal a considerable improvement in ILA efficiency due to ILA-3 [10].

**1.1.4 ILA-4**: The main task of this algorithm is to build a set of classifying rules. It analyzes specific training data with no lost values. The model works iteratively; each repetition explores rules to arrange maximum training samples. Generated rules are acknowledged, and training data samples are labelled. In the next subsequent cycle, such rules are rejected. Especially, ILA requests a rule-per class wherein rule induction unrelated examples in the current category from specimens in the remaining classes. LA-4, designed to work on missing value, builds the best outcome because it holds most problems and impedes other methods. The primary goal of ILA-4 is to grasp the handling and eradication of missing information inside the database. It preserves the importance of critical existing classes while maintaining the integrity of ILA's internal processes, while also maximizing the efficacy of the MCV technique for substituting missing data. The three aforementioned strategies work quite well; the ILA model provides an already processed dataset including all possible permutations of replacement missing values. Missing value management becomes a component of ILA-4's induction phase [2]. ILA-4 adds new features, increasing its ability to handle a wider range of datasets with missing samples, a feature not seen in traditional methods. The compilation time of ILA-4 is a minimal cost that is critical in the structure of data pre-processing.

## 2. Literature Survey

In 2009 Wohlrab et al. collated various approaches to deal with hidden values in datasets. The author also discusses and discriminates general approaches with different strategies: a) Delete strategy: work on hidden values and simply remove all records b) Ignored Value: they are ignored during analysis 3) Any Value: replace hidden values with default values 4) Special Value: replaces hidden values with special values 5) Common Value Strategies: it replaces hidden values with most common values 6) Desperation, Forecast, and Distribution Strategy: This strategy is based on elements such as the value related to despair, predictive calculation, and the method used to dispersing values.

Raja et al. used unsupervised machine learning methods to estimate missing values in 2020. The approach is based on rough set theory, which is a mathematical theory that deals with incomplete information. The author found that their approach achieved favourable results against the UCI benchmark dataset [7].

In 2021, Rashid et al. proposed a model that evaluates the potential benefits of using machine learning for missing value imputation. The model contrasts the accuracy and time efficiency of machine learning-oriented methods with statistics-oriented approaches. The authors also evaluate the efficiency of several MVI (Missing Value Imputation) strategies to existing baseline MVI algorithms such as the k-nearest-neighbour, naive Bayes models, and mode-median methods [8].

In 2021 Liu et al. had; demonstrated MVI techniques that utilize geographical data to close sensors. Data are broadcasted in large volumes; It becomes critical to account for significant data loss across several sensors as a result of a single incident. The author presents a novel approach for Missing Value Imputations (MVI) for univariate time-series data. This method combines an iterative architecture with several segmentation algorithms to efficiently manage significant gaps in the data [11].

In 2018 Do et al. have demonstrated a method to evaluate 30 methods and verify each approach's ability for the following aspects: It builds a map of the biochemical pathways that are involved in association networks. ii) It makes the analysis more powerful while still taking into account the effects of known genes that control metabolism.

In 2018 Mermet et al. have; demonstrated a new concept DRILA. It extends the ILA family algorithm and provides distributed features to access data. It identifies relational rules dispersed across numerous databases connected by logical linkages and disseminated throughout a computer network. This includes data from remote relational databases, a variety of locations, or even a local database for maintenance needs. The resulting rules are critical in anticipating values for unknown entity properties. DRILA uses information from several locations, taking into account any conceivably related schema at each place where linkages between tables are built using foreign keys. The unique hypothesis search technique was used, which avoided the requirement to recomputed rules for improved speed. Notably, the DRILA algorithm processes distributed relational data effectively without requiring a duplicate copy. This feature is critical since it contributes to the algorithm's scalability and usability [5].

In 2021 ravita et al. have; presented Inductive learning in the Job recommendation domain. The author discusses the detailed implementation of all algorithm variations on a different dataset.The algorithm achieve 85 % accuracy in different datasets and algorithms [29]. In 2021 ravita et al. have; presented a new deep semantic model that helps in text mining and semantics representation of the textual entity. The author proposed an Enhanced DSSM algorithm for word embedding [4]. In 2018 ravita et al. presented basic collaborative filtering in job recommendations [12, 15]. In 2021 ravita et al. have; presented the main concepts of Inductive learning and graph-based approach in the job domain. The author proposed a novel graph-based methodology that works on different datasets and improves system efficiency. In 2022 ravita et al. have; presented the advanced Inductive learning for feature selection and missing value removal. The author also highlighted the results of proposed methodology in different datasets. The algorithm achieve 92 % accuracy in different datasets and algorithms [29].

In 2020 ravita et al. presented a Identity resolution approach for Job Recommendation using Collaborative filtering. The model gives best results and 93 % accuracy in different dataset[30].

**Table 2.1:** Overview of Literature Survey

| Author, Year | Contribution, the utilized dataset | Benefits | Research Gap | Accuracy |
|---|---|---|---|---|
| Susan Dumais, 1998 [14] | 1. Compare and contrast the efficacy of five distinct algorithms.<br>2. SVMs are the most precise and time-consuming to train.<br>Dataset: Reuters-21578 (12,902 into 118 categories) | 1. All of the five classifiers are very fast.<br>2. Determine whether a new document should be allocated to a specific category in less than 2 milliseconds. | 1. Few algorithms were considered<br>2. More pre-processing time. | 88%<br>(2MB dataset) |
| [Mehmet R. Tolun, 1998 [3] | 1. ILA-1: 1It generates canonical versions of IF-THEN rules AQ and ID3.<br>2. ILA was created to deal with both discrete and symbolic attributes.<br>3. Stepwise forward method<br>Dataset: Weather dataset | 1. General and robust.<br>2. Rules are suitable for data exploration.<br>3. Concentrate on one rule at a time.<br>4. Overcome attribute selection problem | 1. Dealing with noisy and incomplete samples is not an option.<br>2. Not work with continuous attribute values. | 89.3%<br>(15,000 instances) |
| Oludag M, 1999 [9] | ILA-2: 1. Handles uncertainty in the data.<br>2. ILA-2 handles continuous feature discretization by using the entropy-based algorithm.<br>3. Deals with uncertain data.<br>Dataset: Object classification. | 1. Penalty factor controls performance.<br>2. The hold out method is used to calculate the accuracy of future predictions. 3. Processing time is reduced by using a faster pass criterion. | 1. Feature subset selection give better accuracy.<br>2. The search processing time will be reduced. | 85%<br>(12,000 instances) |
| Saleh M. Abu-Soud,2018 [32] | ILA-3: 1. New Feature Selection Algorithm.<br>2. CombExclude filters out any attribute combinations that aren't relevant.<br>3. It does not take part in the actual | 1. Combats the ID3 and AQ issues<br>2. the new version greatly enhanced the efficiency of ILA | 1. Not handle missing value. | 89.3%<br>(20,000 instances) |

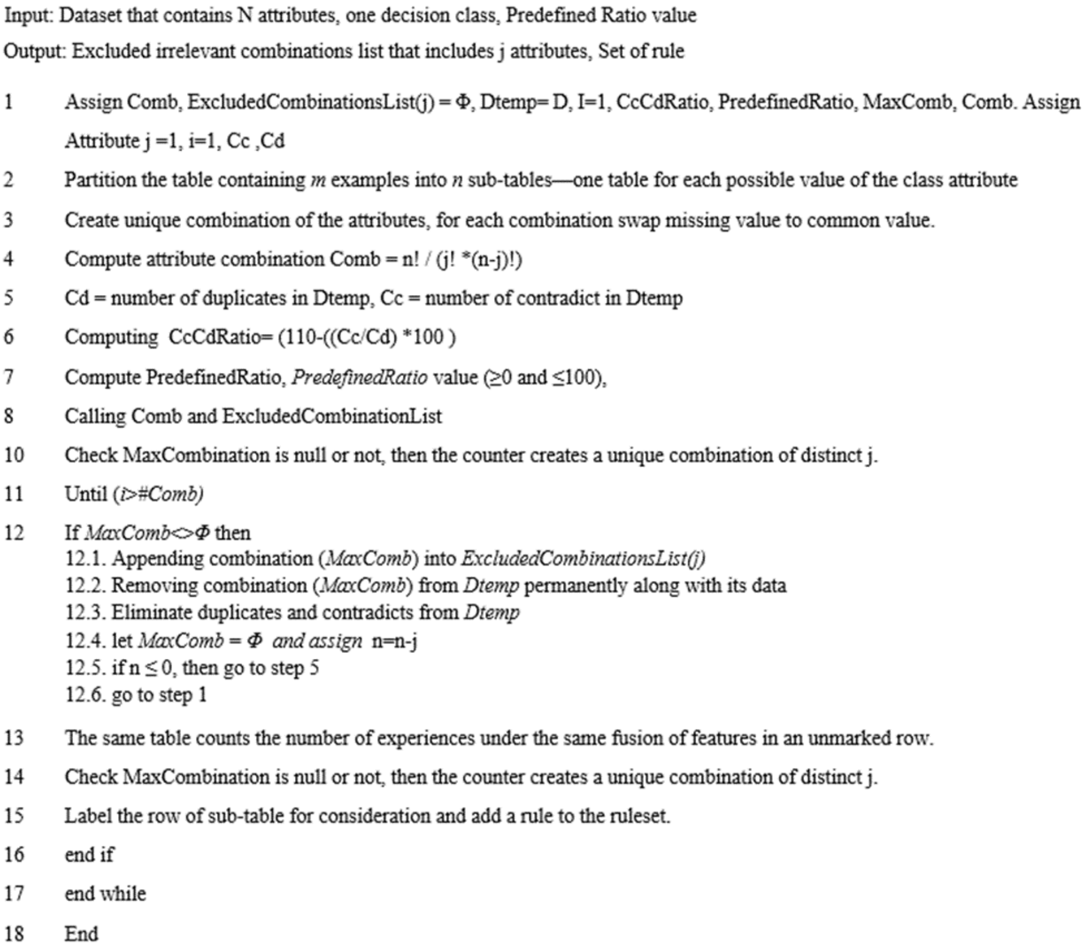
| | | | | |
|---|---|---|---|---|
| | induction procedure.<br>Dataset: Identification of Letters | | | |
| Saleh M. Abu-Soud, 2021 [2] | ILA-4: 1. Misplaced values in datasets can be overcome by machine learning algorithm<br>2. Compared to Regression algorithm, Nave Bayes, and Random Forest, three well-known algorithms<br>3. Better accuracy<br>Dataset: Machine learning repository | 1. When using decision tree techniques, it performs admirably.<br>2. MCV, MCVRC, and Delete strategies are used. | 1. Essential in the outline of the novel approach during the inductive phase is a negligible cost. | 89.6%<br>(1MB dataset) |
| S. et al. 2019 [28] | 1. Modified page Rank algorithm for anomaly detection.<br>2. The configuration utilized ambient sensing modalities.<br>Dataset: MovieLens | 1. Page rank and modified page rank detect the simulated change in the motion pattern. | 1. Standard pattern movements 2. Data collection, Observation is future comparison. | 87.5%<br>(1GB dataset) |
| Ravita et. al 2022[29] | New feature selection technique are used and missing values also handled automatically.<br>Dataset: CareerBuilder dataset and Niti Ayog's dataset | Algorithm work on small and large scale dataset ( centralized and distributed environment) | ----- | 93.5%<br>(5GB Instances) |

## 3. Proposed Methodology

**Problem Statement:** Input is taken from DSSM and graph module. Three graph algorithms are tested on different datasets. Inductive learning-4 (ILA-4) is applied for feature extraction. The results are compared to all inductive learning approaches. The proposed algorithm ILA-5 is applied in the results. It gives more scalable results than existing approaches.

**3.1 Proposed ILA-5 Algorithm (Feature Selection and missing value handling):** The Proposed algorithm ILA-5 is a combination of feature engineering and missing value, and it is used in the pre-processing and post-processing phase of data engineering. This algorithm is mainly helpful in big data applications where data size is enormous and categorical. The above process in the algorithm are designed at once, and they will be called in many places, reducing the code complexity and time. The case study used for this purpose is the CareerBuilder dataset which is not cleaned and pre-processed. The algorithm works at the start of the dataset to format properly and then select the appropriate feature. The Input of this algorithm is a raw CareerBuilder dataset and assigns a few constants like MaxComb, I, J, ExcludedCombinationList etc., and processes the steps iteratively. CareerBuilder and Naukri datasets, but the dataset has a small and medium-size feature, algorithm processing time will increase, but rule generation does not show significant improvement [6].

**Proposed ILA-5 Algorithm**



Input: Dataset that contains N attributes, one decision class, Predefined Ratio value

Output: Excluded irrelevant combinations list that includes j attributes, Set of rule

```
1   Assign Comb, ExcludedCombinationsList(j) = $\Phi$, Dtemp= D, I=1, CcCdRatio, PredefinedRatio, MaxComb, Comb. Assign
    Attribute j =1, i=1, Cc ,Cd
2   Partition the table containing $m$ examples into $n$ sub-tables—one table for each possible value of the class attribute
3   Create unique combination of the attributes, for each combination swap missing value to common value.
4   Compute attribute combination Comb = n! / (j! *(n-j)!)
5   Cd = number of duplicates in Dtemp, Cc = number of contradict in Dtemp
6   Computing  CcCdRatio= (110-((Cc/Cd) *100 )
7   Compute PredefinedRatio, *PredefinedRatio* value ($\geq$0 and $\leq$100),
8   Calling Comb and ExcludedCombinationList
10  Check MaxCombination is null or not, then the counter creates a unique combination of distinct j.
11  Until ($i$>#$Comb$)
12  If $MaxComb$<>$\Phi$ then
    12.1. Appending combination (*MaxComb*) into *ExcludedCombinationsList(j)*
    12.2. Removing combination (*MaxComb*) from *Dtemp* permanently along with its data
    12.3. Eliminate duplicates and contradicts from *Dtemp*
    12.4. let $MaxComb = \Phi$ *and assign* n=n-j
    12.5. if n $\leq$ 0, then go to step 5
    12.6. go to step 1
13  The same table counts the number of experiences under the same fusion of features in an unmarked row.
14  Check MaxCombination is null or not, then the counter creates a unique combination of distinct j.
15  Label the row of sub-table for consideration and add a rule to the ruleset.
16  end if
17  end while
18  End
```

**3.2 Evaluation Measures:** The proposed methodology's performance was evaluated by different measures. There are three main ways to evaluate recommender systems. User studies, online evaluations, and offline evaluations are used to measure the performance of recommender systems and compare different approaches. User studies are conducted on a small scale, with a few dozen or hundreds of people. In a user study, participants are presented with recommendations generated by different recommender algorithms, and then asked to rate the recommendations. Online evaluations are conducted on a larger scale, with thousands or even millions of users. In an online evaluation, two different versions of a recommender system are deployed to different groups of users, and the performance of the two systems is then compared. Offline evaluations are conducted using historical data, such as user ratings or purchase history. In an offline evaluation, a recommender system is trained on historical data, and then its performance is measured on a held-out set of data [2,9].

There are several ways to determine the efficacy of a model, but precision, recall, and accuracy are the most common. These metrics can be calculated by first assessing whether a document was categorized as a true positive (TP), false positive (FP), true negative (TN), or false negative (FN).

**3.3 Confusion matrix:** A confusion matrix is a tabular format that summarizes the performance of a classifier in a binary classification context. The table displays the classifier's counts for true positives (TP), false positives (FP), true negatives (TN), and false negatives (FN). This matrix is useful in determining the efficacy of the classifier. It facilitates in the computation of the following metrics:

TP: Instances successfully categorized as belonging to a specific category.

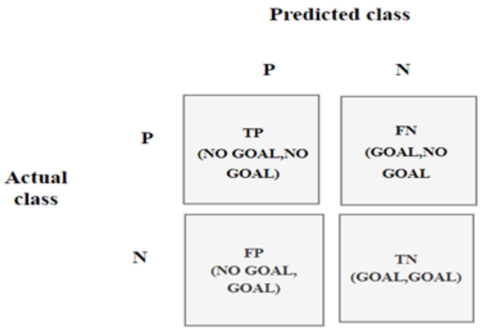
FP: Instances incorrectly classified as belonging to the category.

FN: Instances that should be identified as belonging to a specific category but aren't.

TN: Instances accurately tagged as not belonging to a specific category.

Let's look at football examples to understand better the classification notion in terms of true versus false and positive versus negative. In this example, two definitions are set: the goalkeeper is NO GOAL, a positive class or logic 1, and the goalkeeper is GOAL, a negative class or logic 0. True Positive (TP): An umpire gives a batsman NO GOAL when he is actually NO GOAL. True Negative (TN): When an umpire gives a goalkeeper GOAL when he is actually GOAL. False Positive (FP): This is the condition a goalkeeper is given NO GOAL when he is actually GOAL. False Negative (FN): When an umpire gives a goalkeeper GOAL when he is actually NO GOAL.

The confusion matrix for football is depicted in figure 2 and it help to compute the performance of system.



|  |  | Predicted class |  |
|---|---|---|---|
|  |  | P | N |
| Actual class | P | TP<br>(NO GOAL,NO GOAL) | FN<br>(GOAL,NO GOAL |
|  | N | FP<br>(NO GOAL, GOAL) | TN<br>(GOAL,GOAL) |

**Fig. 2.** Confusion Matrix

Several approaches for determining efficacy, precision, recall, and accuracy are commonly based on the confusion matrix [28].

**Precision:** Precision determines the conditional likelihood that an arbitrary document d is grouped under ci, or what would be deemed the right group. It denotes the classifier's capacity to classify a document as belonging to the proper group rather than all documents belonging to that category, correct or incorrect. It measures the exactness. E.g. the segment of suggested jobs that are suitable. Equation (1) demonstrates the precision of the book recommendation system [67].

$$\boldsymbol{Precision} = \frac{\boldsymbol{tp}}{\boldsymbol{tp+fp}} = | \frac{\boldsymbol{goodbooks\ recommended}}{\boldsymbol{all\ recommendation}} \qquad (1)$$

**Recall:** The likelihood of making this choice if a random document dx should be categorized under the category is denoted as recall (ci). The proportion of genuine positive items recovered over the sum of true positive and false negative items—a fraction of all relevant items retrieved. It assesses the data's completeness. E.g. the segment of all good books suggested. Equation (2) presents the recall of the book recommendation system.

$$\boldsymbol{Recall} = \frac{\boldsymbol{tp}}{\boldsymbol{tp+fn}} = \frac{\boldsymbol{good\ books\ \ recommended}}{\boldsymbol{all\ good\ movies}}$$

**(2)**

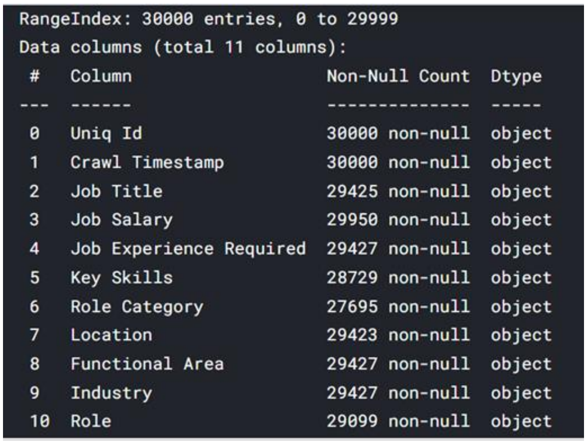
**Accuracy:** Accuracy is frequently used to describe categorization systems. On the other hand, precision and recall values are far more sensitive to changes in the number of accurate decisions. Equation (3) presents the accuracy of the system.

$$\boldsymbol{Accuracy} = \frac{\boldsymbol{tn+tp}}{\boldsymbol{tp+fp+tn+fn}} = \frac{\boldsymbol{Number\ of\ correct\ Prediction}}{\boldsymbol{total\ number\ of\ Prediction}} \qquad \textbf{(3)}$$

**F1-score**: For comparison purposes, the F1-score merges and multiplies the sensitivity and specificity measures into a single value; it achieves a more uniform view of performance and gives equal weight to precision and recall. Because accuracy isn't an appropriate statistic for skewed datasets, precision and recall are used to assess algorithm performance in this scenario. Furthermore, accuracy and recall are frequently combined to visualize the classifier's performance better. This is done by combining them in the next equation (4).

$$\boldsymbol{F1} = \boldsymbol{2}.\frac{\boldsymbol{Precision\ .\ recall}}{\boldsymbol{precision+recall}} \qquad \textbf{(4)}$$

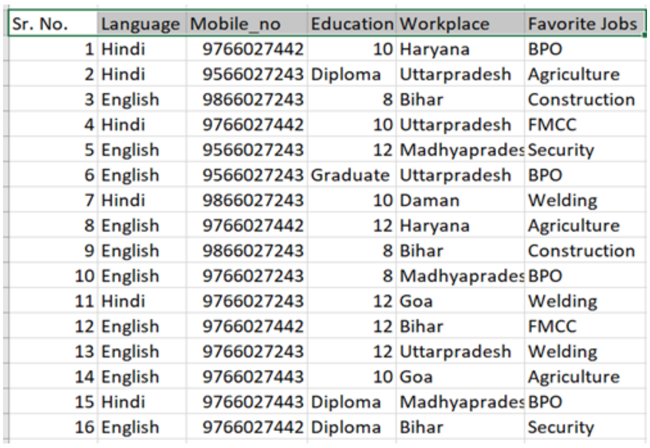
**3.4 Dataset description:** For analysis of ILA-5 algorithm two datasets are used to check the performance of system. Naukri.com job listing dataset: (size 2GB, 234, 000, 00 instances, 38 Features) [24]: Figure 3 and 4 depicts the different features included in datasets. title, walkin, role, view_count, job_type, job_id, vacancy, url, description, industry, company_name, skills, min_salary, max_salary, posted_at, max_experience, min_experience, ug, pg, ppg, locations, currency, salary_label, _id, crawled_at



```
RangeIndex: 30000 entries, 0 to 29999
Data columns (total 11 columns):
 #   Column                    Non-Null Count  Dtype 
---  ------                    --------------  ----- 
 0   Uniq Id                   30000 non-null  object
 1   Crawl Timestamp           30000 non-null  object
 2   Job Title                 29425 non-null  object
 3   Job Salary                29950 non-null  object
 4   Job Experience Required   29427 non-null  object
 5   Key Skills                28729 non-null  object
 6   Role Category             27695 non-null  object
 7   Location                  29423 non-null  object
 8   Functional Area           29427 non-null  object
 9   Industry                  29427 non-null  object
 10  Role                      29099 non-null  object
```

**Fig. 3.** CareerBuilder Job data [23]

**Niti ayog's Job Dataset: (Size 2 KB, 500 instances, 6 Features)** Features: Sr. No, Language, Mobile no, Education, Workplace, Favourite Jobs.



| Sr. No. | Language | Mobile_no | Education | Workplace | Favorite Jobs |
|---|---|---|---|---|---|
| 1 | Hindi | 9766027442 | 10 | Haryana | BPO |
| 2 | Hindi | 9566027243 | Diploma | Uttarpradesh | Agriculture |
| 3 | English | 9866027243 | 8 | Bihar | Construction |
| 4 | Hindi | 9766027442 | 10 | Uttarpradesh | FMCC |
| 5 | English | 9566027243 | 12 | Madhyaprades | Security |
| 6 | English | 9566027243 | Graduate | Uttarpradesh | BPO |
| 7 | Hindi | 9866027243 | 10 | Daman | Welding |
| 8 | English | 9766027442 | 12 | Haryana | Agriculture |
| 9 | English | 9866027243 | 8 | Bihar | Construction |
| 10 | English | 9766027243 | 8 | Madhyaprades | BPO |
| 11 | Hindi | 9766027243 | 12 | Goa | Welding |
| 12 | English | 9766027442 | 12 | Bihar | FMCC |
| 13 | English | 9766027243 | 12 | Uttarpradesh | Welding |
| 14 | English | 9766027443 | 10 | Goa | Agriculture |
| 15 | Hindi | 9766027443 | Diploma | Madhyaprades | BPO |
| 16 | English | 9766027442 | Diploma | Bihar | Security |

**Fig. 4.** Niti Ayog's Job data [25]

## 4. Results and Discussion

A random selection of 10-15% of job data is used to evaluate the effectiveness of the top-k recommendations and their related rankings in ILA-5. This evaluation includes two metrics: Precision@k and Recall@k. The value of k is normally between 10 and 30, and interim results are published for all users in the test set. The evaluation technique entails examining six matched tasks that include click, app, and preference graphs. The Performance analysis of ILA-5 algorithm on embedding model system. We consider 5 baseline approaches to check the algorithm's efficiency for evaluation purposes. GBA (graph-based approach), Joint BPR and Margin, SJRS (Scalable job recommendation system), ILA-5 (Inductive learning approach-5 [17]).

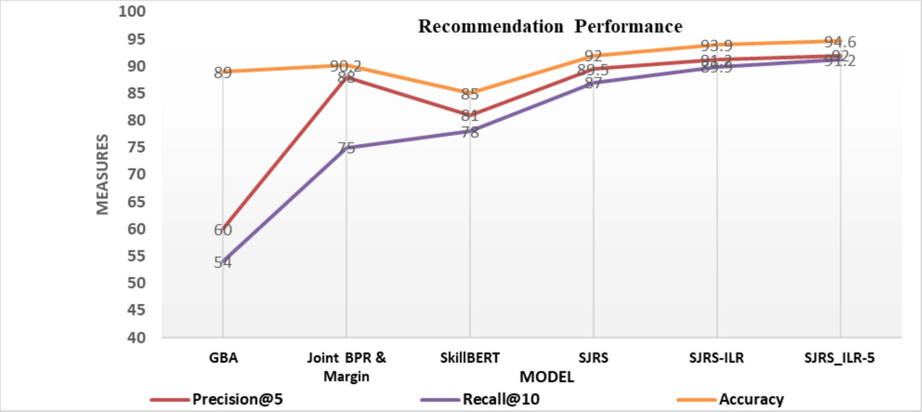


**Fig. 5.** Recommendation Performance on ILA-5

The above graph shows the precision or correctness of the basic and advanced and proposed E-DSSM model. Model LDA precision is low compared to other models, but BERT, SkillBERT and Graph embedding performance increases. As we observe, the model E-DSSM gives better performance than other models.

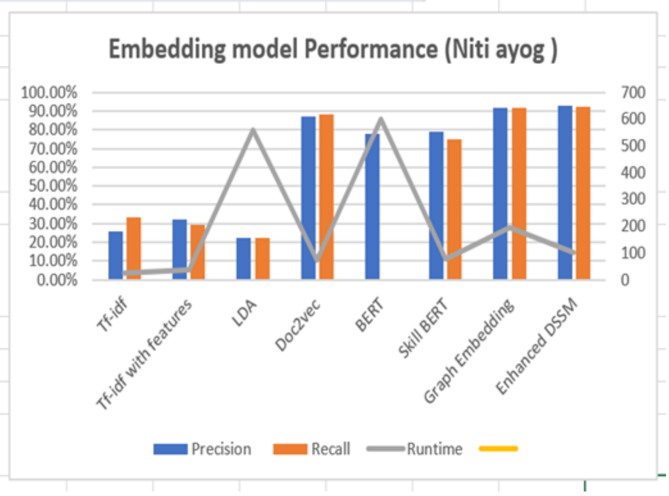


Fig. 6. Model Performance on NITI Ayog’s dataset [25]

The above graph shows the recall of the basic and advanced, and proposed E-DSSM model. Model LDA decreases the recall value, and doc2vec has a higher recall than other models; graph embedding performance increases. As we observe, the model E-DSSM gives better performance than other models. Figure 7 shows run-time of the basic, advanced, and proposed E-DSSM model.

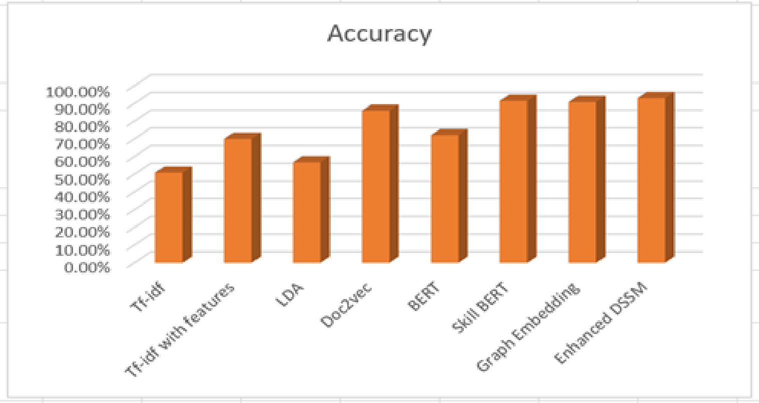


**Fig. 7.** Accuracy of Model

the Model LDA takes more time to compare to other models. As we observe, the model E-DSSM takes moderate time to process the results.

### 4.1 Analysis:

The precision of the embedding model is demonstrated by the findings (tf-idf, LDA, doc2vec, BERT, SkillBERT [19], Graph Embedding and Enhanced DSSM). The results demonstrate that the Niti ayog's job portal dataset contains fewer features than other datasets, but the skill and job descriptions are different and do not match our OCE database. The embedding model E-DSSM provides 93.0 percent skill and job matching for these datasets, and the system's run-time performance is around 100ms. The model's accuracy is also said to be good.

To improve the performance inductive learning approach with new variations will be applicable. ILA model outcomes are shown below. The rule creation and missing value handling will only occur in the first phase using the SGBA-ILA algorithm [26, 27] and running it in a PySpark cluster with eight nodes (cluster).

As a result, the model executes the decision class value during the testing phase. The model's run time was 75 milliseconds with a 93.9 percent accuracy.

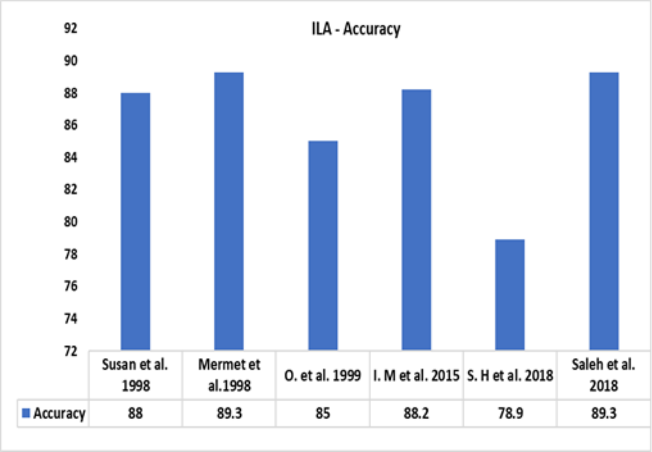


Fig. 8. ILA Accuracy

Figure 8 shows the accuracy of the inductive learning algorithms, and they are widely used in different domains. The algorithm ILA-3 feature selection and ILA-3 missing value handling will have the best accuracy and be used in the big data domain. Other existing algorithms, ILA-1 and ILA-2, are not suitable for big data and diverse features—the ILA-3 and ILA-4 considering as our baseline algorithms for performance evaluation.

**TABLE 2.** Comparison with baseline approach [4]

| **Metrics** | **GBA** | **Joint BPR & Margin** | **SkillBERT** | **SJRS** | **SJRS_ILR-5** |
|---|---|---|---|---|---|
| Precision@5 | 60% | 88% | 81% | 89.50% | 92 |
| Recall@10 | 54% | 75% | 78% | 87% | 91.2 |
| Accuracy | 89% | 91.24% | 85% | 92.00% | 94.6 |

### 4.2 Verification and validation

**Verification**: We refer four methodology for result comparison and validation purposes. Four dataset also selected for result validation purpose and compare their methodology, embedding model, and contributions and performance ( precision, recall and F1-score). The four methodology graph based approach, Joint BPR & margin model, Skillbert approach, Scalable Job Recommendation System are used for comparison purpose. The results clearly show that the SJRS with inductive learning performs better and satisfactory results. Hence the proposed graph-based model with inductive learning is preferred to improve the system scalability. Table 2 compares the acceptance of the model developed with the reference model [16, 18]. It shows that the responsivity is comparable to the reference model [69] for similar precision and accuracy.

The proposed model's performance is measured using appropriate parameters in different datasets and satisfactory results.

## 5. Conclusion

The job recommendation system has a number of obstacles, most notably scalability and performance. To improve the system's scalability, an algorithm capable of managing large and heterogeneous datasets is required. We provide a new strategy that combines machine learning techniques with the substitution of various concealed values inside datasets. In this quest, we use the ILA inductive learning algorithm from previous research to demonstrate its inductive capabilities. We predict an improvement in system performance by enhancing our technique with new features and broadening its reach to accept datasets with missing instances. The model's preliminary evaluations and comparison studies reveal a higher degree of accuracy. For dealing with hidden values, the model employs well-known strategies such as the Most Common Value (MCV), MCV with Random Choice (MCVRC), and Deletion strategy. Empirical analysis reveals that our suggested technique produces remarkable and beneficial results, particularly in terms of the amount and complexity of the created rules. Following the implementation of the updated ILA, the cost of adopting the new process during inductive learning is insignificant. The efficacy of the proposed model is evaluated using important parameters across several datasets, providing good results.

### Conflicts of interest

The authors declare no conflicts of interest.

## References

[1] J.R. Quinlan. Learning efficient classification procedures and their application to chess end games, in: R.S. Michalski, J.G. Carbonell and T.M. Mitchell, eds., Machine Learning: An Artificial Intelligence Approach (Morgan Kaufmann, San Mateo, CA, 1983).

[2] Ammar Elhassan, Saleh M. Abu-Soud, Firas Alghanim, Walid Salameh, "ILA4: Overcoming missing values in machine learning datasets – An inductive learning approach", Journal of King Saud University – Computer and Information Sciences xxx (XXXX) xxx

[3] Mehmet R. Tolun, Saleh M. Abu-Soud, "ILA: an inductive learning algorithm for rule extraction", Expert Systems with Applications Volume 14, Issue 3, April 1998, Pages 361-370, https://doi.org/10.1016/S0957-4174(97)00089-4.

[4] Ravita Mishra, Sheetal Rathi, Enhanced DSSM (Deep Semantic Structure Modelling) Technique for Job Recommendation, Journal of King Saud University - Computer and Information Sciences, 2021, ISSN 1319-1578, https://doi.org/10.1016/j.jksuci.2021.07.018.

[5] Abu-Soud S. and Al Ibrahim A., DRILA: A Distributed Relational Inductive Learning Algorithm, WSEAS Transactions on Computers, Issue 6, Volume 8, June 2009, ISSN: 1109-2750.

[6] Oludag M., Tolun M., Sever H., and Abu-Soud S., "ILA-2: An Inductive Learning Algorithm for Knowledge Discovery", Cybernetics and Systems: An International Journal, vol. 30, no. 7, Oct.-Nov. 1999.

[7] Raja, P.S., Thangavel, K. "Missing value imputation using unsupervised machine learning techniques", Soft Comput. 24, 4361–4392. 2020, https://doi.org/ 10.1007/s00500-019-04199-6.

[8] Rashid W., Gupt, M.K., "A Perspective of Missing Value Imputation Approaches". In: Gao, X. Z., Tiwari, S., Trivedi, M., Mishra, K. (eds) Advances in Computational Intelligence and Communication Technology. Advances in Intelligent Systems and Computing, vol 1086. Springer, Singapore.2021, 10.1007/ 978-981-15-1275-9_25.

[9] Xingdog WU, "Inductive Learning: Algorithms and Frontiers" Department of Artificial Intelligence, University of Edinburgh, 80 South Bridge, Edinburgh EH1 1HN, UK, Artificial intelligence Review 7.

[10] Saleh M. Abu-Soud and Sufyan Almajali, "ILA-3: An Inductive Learning Algorithm with a New Feature Selection Approach ", WSEAS Transactions on Systems and Control · January 2018.

[11] Le Wu, Yonghui Yang, Lei Chen, Defu Lian, Richang Hong, Meng Wang, "Learning to Transfer Graph Embeddings for Inductive Graph-based Recommendation ", SIGIR '20, July 25–30, 2020, Virtual Event, China, https://doi.org/10.1145/3397271.3401145.

[12] Ravita Mishra, Dr Sheetal Rathi, "Efficient and Scalable Job Recommender System Using Collaborative Filtering", Paprzycki M., Gunjan V. (eds) ICDSMLA 2019. Lecture Notes in Electrical Engineering, vol 601. Springer, Singapore https://doi.org/10.1007/978-981-15-1420-3_91.

[13] George V. Lashkia, Laurence Anthony, "An inductive learning method for medical diagnosis", Pattern Recognition Letters 24 (2003) 273–282, Received 30 October 2001; received in revised form 23 April 2002.

[14] Susan Dumais, John Platt, David Heckerman, Mehran Sahami, "Inductive Learning Algorithms and Representations for Text Categorization", Proceedings of the seventh international conference on information and knowledge management. ACM.1998.

[15] Shuai Zhang, Lina Yao, Aixin Sun, and Yi Tay. 2018. Deep Learning based Recommender System: A Survey and New Perspectives. ACM Comput. Surv. 1, 1, Article 1 (July 2018), 35 pages. DOI: 0000001.0000001.

[16] Vachik S. Dave, Baichuan Zhang, Mohammad AI Hasan, Khalifeh Aljadda and Mohammad Korayem, "A combined representation learning approach for better job and skill recommendation", CIKM '18 ACM ISBN 978-1-4503-60149-2/18/10. DOI: 10.1145/3269206.3272023., ACM-2018.

[17] Pavlos Kefalas, Panagiotis Symeonidis, and Yannis Manolopoulos, "A Graph-Based Taxonomy of Recommendation Algorithms and Systems in LBSNs", IEEE transaction on knowledge and data engineering, Vol. 28, NO. 3, March 2016.

[18] Amber Nigam, Shikha Tyagi, Kuldeep Tyagi, Arpan Saxena, "SkillBERT: "Skilling "the BERT to classify skills!", ICLR 2021 Conference.

[19] Charu C aggrawal, "Recommender system Textbook", ISBN 978-3-319-29657- 9 ISBN 978-3-319-29659-3, DOI 10.1007/978-3-319-29659-3, Springer International Publishing Switzerland 2016.

[20] Vedant Bhatia, P Rawat, A Kumar, RR Shah, "End-to-End Resume Parsing and Finding Candidates for a Job Description using BERT", arXiv preprint arXiv:1910.03089, Computer Science, Information Retrieval, 2018.

[21] Luca G. Cellamare_, Michele A. Bertoldi_, Alberto

Parravicini†, Marco D. Santambrogio," Exploring transductive and inductive methods for vertex embedding in biological networks", 978-1-7281-3815-2/19/$31.00 ©2019 IEEE.

[22] www.careerbuilder.com

[23] www.naukri.com

[24] http://www.niti.gov.in

[25] Ravita Mishra, Sheetal Rathi," Inductive Learning in Job Recommendation", International Journal of Intelligent Systems and application in Engineering, ISSN: 2147 679.

[26] Ravita Mishra, Sheetal Rathi, "Scalable graph-based approach (SGBA) in Job recommendation system", Springer Journal of Soft Computing. (Springer Publisher), SOCO-D-21-04145, Nov 2021 (Under Review).

[27] Adrien Mogenet, Tuan-Anh Nguyen Pham, Masahiro Kazama, Jialin Kong. 2019. Predicting Online Performance of Job Recommender Systems with Offline Evaluation. In Thirteenth ACM Conference on Recommender Systems (RecSys' 19), September 16–20, 2019, Copenhagen, Denmark. ACM, New York, NY, USA, four pages. https://doi.org/10.1145/3298689.3347032.

[28] R. Ravita and S. Rathi, "Inductive Learning Approach in Job Recommendation", Int J Intell Syst Appl Eng, vol. 10, no. 2, pp. 242–251, May 2022.

[29] Pandey, Mayuresh, and Ravita Mishra. "Identity Resolution In Social Network Using Recommender System." In e-Conference on Data Science and Intelligent Computing, p. 97. 2020.

[30] Dr. S.A. Sivakumar. (2019). Hybrid Design and RF Planning for 4G networks using Cell Prioritization Scheme. International Journal of New Practices in Management and Engineering, 8(02), 08 - 15. https://doi.org/10.17762/ijnpme.v8i02.76

[31] Bamber, S. S. . (2023). Evaluating Performance of Beacon Enabled 802.15.4 Network with Different Bit Error Rate and Power Models. International Journal on Recent and Innovation Trends in Computing and Communication, 11(2s), 167–178. https://doi.org/10.17762/ijritcc.v11i2s.6040

[32] Jain, V., Beram, S. M., Talukdar, V., Patil, T., Dhabliya, D., & Gupta, A. (2022). Accuracy enhancement in machine learning during blockchain based transaction classification. Paper presented at the PDGC 2022 - 2022 7th International Conference on Parallel, Distributed and Grid Computing, 536-540. doi:10.1109/PDGC56933.2022.10053213 Retrieved from www.scopus.com