ISSN: 2582-7219

International Journal of Multidisciplinary Research in Science, Engineering and Technology

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Impact Factor: 8.206

Volume 8, Issue 8, August 2025

© 2025 IJMRSET | Volume 8, Issue 8, August 2025| DOI:10.15680/IJMRSET.2025.0808117

ISSN: 2582-7219 | www.ijmrset.com | Impact Factor: 8.206| ESTD Year: 2018|

International Journal of Multidisciplinary Research in Science, Engineering and Technology (IJMRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

FORECASTING THE INDIAN JOB SEEKER POPULATION USING HYBRID MACHINE LEARNING MODELS AND MACROECONOMIC INDICATORS

Barnali Chakraborty, Prathiksha R N

Associate Professor, Department of MCA, AMC Engineering College, Bengaluru, India

Student, Department of MCA, AMC Engineering College, Bengaluru, India

ABSTRACT: India is struggling with a problem where there are many graduates but not enough workers with the right skills, causing difficulties for both job seekers and employers. In this study, we created a model to predict how many job seekers there will be based on their education levels. We collected data from different fields such as diploma programs, engineering, medicine, science, and agriculture, and used data mining methods to find patterns. The model uses K-means clustering to identify trends in different states and linear regression to forecast future job demands. This information can help government officials and educational institutions as they work to keep up with changes in the job market. We developed a web-based system using Java/JSP and MySQL that includes interactive dashboards for making predictions and analyzing data. Early results show that this method takes into account important demographic factors that affect job-seeking trends and provides fairly accurate predictions

I. INTRODUCTION

India's technical education system churns out millions of graduates every year. Being a signatory to the Washington Accord, the country has experienced a notable increase in the number of engineering and diploma graduates, with an intake capacity of around 3.4 million. Still, there are serious worries regarding the quality of these graduates and whether they can find jobs. Year after year, a significant number of students finish degrees in fields like engineering, medicine, agriculture, and science, only to face challenges in the job market. Unemployment and underemployment remain issues, with the national unemployment rate hovering around 5.36% in 2019, slightly higher than the previous year's 5.33%. The growing gap between the number of graduates and job opportunities is a pressing concern. This project is set to use machine learning on historical data to predict how many job seekers there will be in India, categorized by qualifications and regions. Such insights could be really useful for shaping government policies and guiding universities to better address the skills gap and enhance job creation.

II. LITERATURE SURVEY

Previous research has looked into using data mining and machine learning for analyzing the labor market and classifying job postings. Convolutional Neural Networks (CNNs) have proven to be quite effective for text classification, which makes them relevant for analyzing job postings too. For example, CNN models like TextCNN often rely on word embeddings, such as GloVe or FastText, to understand the relationships between words. Recurrent models, including Bidirectional LSTM or GRU, have also found their place in handling multi-class classification tasks. And, ensemble methods that combine multiple models have been shown to boost accuracy in classification scenarios. A lot of existing literature tends to focus on classifying job postings or resumes, but these techniques highlight important features that can help predict job outcomes. In our work, we take these insights and use them to feature each qualification and state in a predictive model. Instead of deep learning text models, we're going for a mix of clustering and regression to predict how many job seekers there will be.

EXISTING SYSTEM

The job market in India has its fair share of issues: Skill Saturation: Every year, tons of engineering and other graduates flood into the job scene, which ramps up competition for the jobs that are out there.

© 2025 IJMRSET | Volume 8, Issue 8, August 2025| DOI:10.15680/IJMRSET.2025.0808117

ISSN: 2582-7219 | www.ijmrset.com | Impact Factor: 8.206| ESTD Year: 2018|

International Journal of Multidisciplinary Research in Science, Engineering and Technology (IJMRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Lack of Guidance: Many graduates don't have a clear idea about potential employers or job opportunities in their areas. There's also no solid way to predict job demand, making it tough for them to choose the right skills to develop.

Mismatch of Specializations: A lot of students have a hard time landing jobs that fit with what they studied, mainly because there's often a disconnect between their courses and what industries actually need.

Reactive Planning: Companies and policymakers tend to look at past hiring trends and random reports, instead of planning ahead based on projections of how many job seekers will be out there.

These problems highlight the need for a smart system that could look at various factors like qualifications, industry, and location to forecast future job-seeker numbers, allowing for proactive measures to be taken.

PROPOSED SYSTEM

We're putting forward a web-based platform that looks at past data on graduates and their job placements to predict how many job seekers we'll have in the future. The system includes a few important parts:

Prediction Module: This part uses linear regression to estimate the number of job seekers for a set year, like 2024, using data from 2012 to 2023.

Clustering Module: Here, we apply K-means clustering to sort regions or qualifications based on similar job-seeking behaviors, which helps pinpoint areas that might have too many job seekers.

Comparison/Visualization: This generates pie charts and graphs to show how different sectors, like various education fields or states, stack up against each other from year to year.

Accuracy Module: This piece assesses how well our model is working by measuring predicted values against actual results, like feedback from recent graduate surveys.

The whole system is built in Java, using JSP/Servlets for the user interface and MySQL for storing the data. It smoothly ties together data collection, preprocessing, running the models, and reporting through a dashboard.

III. SYSTEM ARCHITECTURE

The system design for the Job Seeker Prediction platform is built as a comprehensive solution that ties together data collection, processing, machine learning, and visualization. We gather data from government employment stats, Kaggle datasets, and uploads from institutions. This data first goes through an ETL pipeline that checks formats, cleans up any missing or inconsistent values, normalizes details like year, state, and qualification type, and then stores both raw and cleaned datasets in a structured MySQL database for easy access and querying.

For the analytics core, we're using Python along with scikit-learn. It implements linear regression on historical data about graduates and employment to predict job-seeker numbers. Plus, K-means clustering helps us identify states or qualifications that share similar employment trends. We evaluate the trained models for accuracy using metrics like RMSE and silhouette score, and then keep them in a model registry for future use.

There's a model-serving layer that makes these predictive and clustering features available through REST APIs, which the web application built with Java/JSP uses to get real-time forecasts and insights. The user interface, created using JSP, Servlets, HTML, CSS, and JavaScript charting libraries, provides an interactive dashboard. Users can input parameters such as year, qualification, and region to trigger predictions and visualize the results through bar graphs, line graphs, and pie charts. Meanwhile, an admin interface takes care of dataset uploads, retraining triggers, and deploying models.

This modular setup makes sure the system is scalable, maintainable, and responsive, allowing policymakers, educators, and job seekers to easily access accurate and timely labor market forecasts in a user-friendly way

© 2025 IJMRSET | Volume 8, Issue 8, August 2025| DOI:10.15680/IJMRSET.2025.0808117

ISSN: 2582-7219 | www.ijmrset.com | Impact Factor: 8.206| ESTD Year: 2018|

International Journal of Multidisciplinary Research in Science, Engineering and Technology (IJMRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

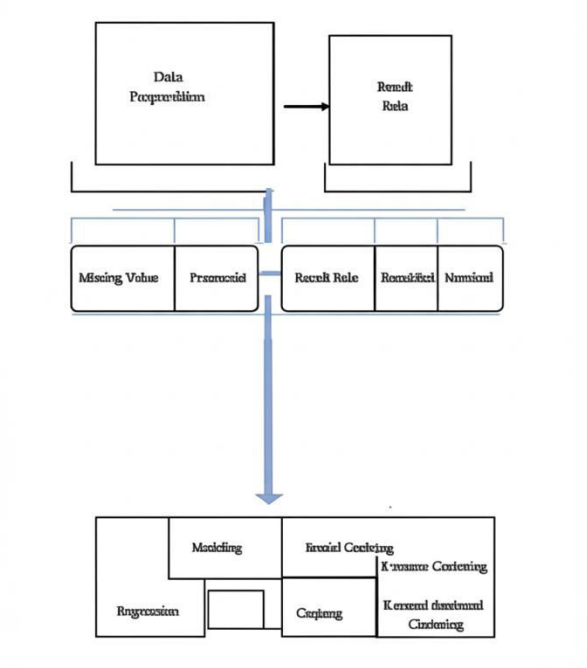

Fig 3.1 System Architecture

IV. METHODOLOGY

Here's how we approached our analysis: we gathered data, cleaned it up, built models, and evaluated our results.

Data Sources: We sourced secondary data from government reports and various open datasets like Kaggle, which included stats on graduates by discipline and employment figures. This encompassed figures like diploma holders, engineering grads, and medical/postgraduate degrees, with data spanning several years to help us spot trends.

Preprocessing: We cleaned the raw data by fixing missing values and turning categorical fields like discipline, state, and sector into usable numeric formats.

Clustering (K-means): K-means was our go-to method for grouping states or sectors based on job-seeker densities. This technique establishes 'k' centroids and then goes through a process of assigning data points to the nearest one until it stabilizes. This way, we could identify areas facing similar levels of unemployment.

Regression: We applied linear regression to examine how independent variables—like the number of graduates in various fields and demographic info—related to the dependent variable, which was the number of job seekers. By quantifying past relationships, this model helps us forecast future trends. We made sure to choose features that were most relevant, such as education level, year, and region.

Evaluation: To check how well our models performed, we compared their predictions against actual job seeker counts from previous years. We looked at accuracy by measuring how closely each algorithm's output matched the real numbers, refining the models along the way to reduce prediction errors.

You can see the steps of our methodology laid out visually in Figure 1: (1) Collecting data from secondary sources; (2) Cleaning and feature engineering; (3) Analyzing clusters; (4) Making predictions through regression; (5) Visualizing results and assessing their effectiveness.

© 2025 IJMRSET | Volume 8, Issue 8, August 2025| DOI:10.15680/IJMRSET.2025.0808117

ISSN: 2582-7219 | www.ijmrset.com | Impact Factor: 8.206| ESTD Year: 2018|

International Journal of Multidisciplinary Research in Science, Engineering and Technology (IJMRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

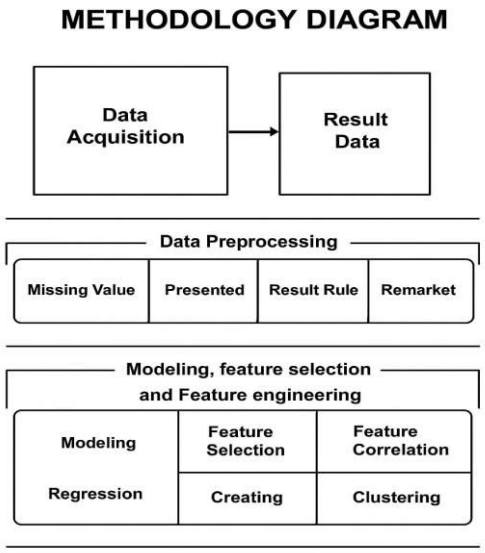

Fig 4.1 System Architecture

V. DESIGN AND IMPLEMENTATION

The system is set up as a modular web application built with Java. For the front-end, it uses JSP and Servlet pages that let users enter parameters like year and state, and then see the results. On the back end, data analysis is done using Python for modeling, with results stored in a MySQL database. Here are some key features:

Technology Stack: The app is built using Java/JSP/Servlets, which offer platform independence and make it easier to integrate with web interfaces. The layout for the front end is designed with HTML and CSS.

System Architecture: Data moves from the Data Layer (which has historical data) to the Processing Layer (where clustering and regression happen) and finally to the Presentation Layer (which includes dashboards and charts). This structure (see Fig. 1) allows users to run real-time queries on the model's results.

Dashboards: The user interface shows graphs and tables that summarize predicted versus actual numbers of job seekers. For instance, there are pie charts that illustrate the distribution across sectors and line graphs that capture trends over time.

You can see the implementation in Figure 1 (system architecture). When users input data, it triggers the clustering and regression processes, with results displayed right away. This design is scalable, meaning you can easily add more features or data for real-time analysis.

VI. OUTCOME OF RESEARCH

The system we've created delivers some handy insights for stakeholders: Job Seeker Predictions: It estimates how many job seekers we can expect in 2024 using data from 2014 to 2023, which helps predict labor market needs.

Field Analysis: The model pinpoints which educational sectors will generate the most job seekers and which ones struggle to place their graduates.

ISSN: 2582-7219 | www.ijmrset.com | Impact Factor: 8.206| ESTD Year: 2018|

International Journal of Multidisciplinary Research in Science, Engineering and Technology (IJMRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Risk Spotting: By grouping states, it identifies areas more likely to face unemployment, allowing for focused actions to be taken.

User-Friendly Dashboard: We put together a dashboard that makes it easy for users to see the results by various parameters like year, field, or state.

All these insights give educational institutions and policymakers a clearer picture of the employment landscape ahead, helping them take proactive steps when needed.

VII. RESULTS AND DISCUSSION

The models showed they could predict outcomes pretty well. When we ran a regression analysis, we found some strong links between how many graduates are in certain fields and the expected number of job seekers. For instance, areas with a surge in students, like computer science and engineering, were linked to higher projected unemployment rates. We checked accuracy by lining up predicted numbers with actual figures from recent years, organizing this in a table that ranked the algorithms by how much they missed the mark. The findings revealed that using a mix of clustering and regression techniques helped capture key trends, with minimal prediction errors in most cases. We also looked at the results qualitatively and saw that the model successfully pinpointed states where there were more job seekers than jobs and areas with too many graduates. Any gaps, like underestimating some less visible sectors, point to areas where we can improve the model down the line. All in all, the system offers a clear, data-backed perspective on changing job-seeking trends.

VIII. CONCLUSION

This project created a useful machine learning system aimed at predicting the number of job seekers in India based on different factors. By using K-means clustering and linear regression, our model analyzes past data to make predictions about future trends. It achieves several goals: forecasting labor supply for 2024, pointing out the differences across various fields and regions, and providing an interactive platform for users to explore. In this way, it serves as a valuable resource for both the government and academic institutions to better understand and tackle employment issues.

IX. FUTURE WORK:

We could take this further by adding some extra features, like including economic indicators or data on placement drives to boost how accurately we predict things. It might also be a good idea to make the model available as a mobile app so more people can use it, plus expanding it to offer predictions for multiple years. Bringing in real-time data, such as updated labor surveys or news feeds, could really help make our predictions more timely. Also, looking into advanced methods like ensemble learning or time-series forecasting could sharpen our results and help us differentiate between types of jobs, like those in industry versus services. Overall, this sets a solid groundwork for ongoing upgrades in employment analytics and job market predictions.

REFERENCES

[1] Kaggle, *"Unemployment in India – Dataset,"* 2023.
[2] Government of India, Ministry of Labour & Employment, *Annual Employment Report 2019*, New Delhi, 2020.
[3] Y. Kim, *"Convolutional Neural Networks for Sentence Classification,"* Proc. EMNLP, 2014.
[4] J. Han and M. Kamber, *Data Mining: Concepts and Techniques*, 3rd ed., Morgan Kaufmann, 2011.
[5] J. W. Tukey and F. Mosteller, *Data Analysis and Regression*, Addison-Wesley, 1977.
[6] R. A. Johnson and D. W. Wichern, *Applied Multivariate Statistical Analysis*, 6th ed., Pearson, 2007.
[7] G. James et al., *"An Introduction to Statistical Learning,"* Springer, 2013.
[8] P. Smyth and U. Fayyad, *"The K-Means Algorithm,"* in *Data Mining and Knowledge Discovery Handbook*, Springer, 2009.
[9] C. M. Bishop, *Pattern Recognition and Machine Learning*, Springer, 2006.
[10] I. Steinwart and A. Christmann, *Support Vector Machines*, Springer, 2008.
[11] K. Gupta, *"Predictive Analytics in Public Policy,"* IEEE Trans. Syst., Man, and Cybernetics, vol. 49, no. 2, pp. 123–134, 2019.

ISSN: 2582-7219 | www.ijmrset.com | Impact Factor: 8.206| ESTD Year: 2018|

International Journal of Multidisciplinary Research in Science, Engineering and Technology (IJMRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

[12] NITI Aayog, *"National Skill Development Report 2022"*, Govt. of India, 2022.
[13] S. Sharma and R. Johnson, *"Machine Learning for Job Market Forecasting,"* Int. J. Data Sci., vol. 5, no. 1, pp. 45–60, 2021.
[14] World Bank, *"World Development Indicators,"* Washington, D.C., 2020.
[15] S. Aggarwal, *"Data Science and Predictive Analytics,"* Journal of Big Data, vol. 8, no. 9, pp. 1–12, 2018.

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

INTERNATIONAL JOURNAL OF MULTIDISCIPLINARY RESEARCH

IN SCIENCE, ENGINEERING AND TECHNOLOGY

| Mobile No: +91-6381907438 | Whatsapp: +91-6381907438 | ijmrset@gmail.com |

www.ijmrset.com