Journal of Theoretical and Applied Information Technology
15th March 2025. Vol.103. No.5

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

LEVERAGING COUNT VECTORIZER FOR JOB TITLE PREDICTION: A COMPARATIVE STUDY OF MACHINE LEARNING ALGORITHMS

D.DEEPTHI[1], KAMINENI B.T.SUNDARI[2], B.SATISH BABU[3], K.THRILOCHANA[4], A.MAHALAKSHMI[1], SNEHA.H.DHORIA[5*]

[1] Assistant Professor, R.V.R. & J.C. College of Engineering, Guntur, Department of Computer Science and Business Systems, Guntur, India.
[2] Sr.Assistant Professor, Department of Computer Science Engineering -Cyber Security, Geetanjali College of Engineering and Technology, Medchal.
[3]Assistant Professor, R.V.R. & J.C. College of Engineering, Guntur, Department of Information Technology, Guntur, India.
[4]Assistant Professor, Vasireddy Venkatadri Institute of Technology, Department of Information Technology, Guntur, India.
[5]Assistant Professor, R.V.R. & J.C. College of Engineering, Department of Mechanical Engineering, Guntur, India.
Corresponding E-Mail:10.deepthi@gmail.com

ABSTRACT

Job title prediction from description data is a crucial task in automating job classification and improving digital job search platforms. This study evaluates the performance of advanced machine learning models—K-Nearest Neighbors, Support Vector Machines, Gradient Boosting, and Logistic Regression—for predicting job titles based on descriptive text. Experiments were conducted using two data splits: 70-30 and 80-20 for training and testing. Results reveal that the 70-30 split consistently outperforms the 80-20 configuration in terms of prediction accuracy. Among the evaluated models, Gradient Boosting achieved the highest performance, with an accuracy of 98.05%, utilizing the Count Vectorizer method. Furthermore, Gradient Boosting recorded the highest F1-score of 0.88, along with a recall of 0.81 for the class for 70-30. These findings highlight the superior capability of Gradient Boosting in capturing complex patterns in textual data and emphasize the significance of data pre-processing and splitting strategies. The outcomes contribute to the optimization of machine learning applications in employment platforms, enhancing user experience and efficiency in matching candidates with appropriate job opportunities. This paper focuses on the gap in bringing job seekers and proper opportunities to improve a more transparent, efficient, and trustworthy job marketplace.
Keywords: *Job title, Machine learning, Job descriptions, Gradient Boosting, Count Vectorizer, F1-score.*

I. INTRODUCTION

In the dynamic and fast-paced online job market, the proliferation of job postings has created dual-edged circumstances. On one hand, it opens up an extensive range of opportunities for job seekers, providing access to roles across industries, geographies, and specializations. On the other hand, it introduces challenges that significantly complicate the job search process. [1] Among these challenges, the overwhelming volume of job postings often obscures the visibility of relevant opportunities, leaving job seekers to sift through countless listings. [2] This is further exacerbated by inaccuracies in job titles and vague descriptions, which fail to convey the precise nature of the roles being advertised. For instance, job titles like "Engineer" or "Analyst" may appear across various domains such as software development, mechanical engineering, or business analytics, leading to ambiguity. Similarly, titles such as "Rockstar Developer" or "Sales Ninja" add to the confusion by employing unconventional and non-standard terminology. [3] As a result, job seekers may either overlook roles that match their qualifications or waste time exploring irrelevant listings, thereby reducing the efficiency of the job search process. This issue is particularly pressing in today's context, where the accurate alignment of candidates with job roles is critical not only for individual career growth but also for organizational productivity. Hiring mismatches can lead to wasted

resources, decreased employee satisfaction, and reduced operational efficiency. As the stakes continue to rise, it becomes imperative to develop advanced, data-driven techniques capable of navigating the complexities of job descriptions and providing accurate predictions of job titles.

Recent advances in machine learning (ML) have demonstrated the potential to revolutionize this domain. By leveraging sophisticated algorithms and feature extraction techniques, these methods can process vast amounts of textual data, identify patterns, and make precise predictions. Machine learning models such as Support Vector Machines, Gradient Boosting, and Logistic Regression are particularly well-suited for such tasks, as they excel in handling high-dimensional data and learning intricate relationships between input features. Thus, addressing the challenges of the online job market requires robust and scalable solutions that can accurately parse and analyze job descriptions. These solutions not only improve the relevance and precision of job recommendations but also contribute to a more transparent, efficient, and equitable job marketplace.

Machine learning has proven to be a critical instrument for the complexity of the challenge in job title prediction from job descriptions [4]. Unlike traditional recruitment methods, which often depend on manual classification or simplistic keyword matching, machine learning algorithms can reveal more profound patterns and relationships within data. Techniques such as K-Nearest Neighbours (KNN), Support Vector Machines (SVM), Gradient Boosting, and Logistic Regression are highly valued for their ability to interpret the semantic nuances of job descriptions and map them to appropriate job titles. Such approaches, which not only automate classification processes but also ensure a very high degree of accuracy and reliability, address a crucial gap in the digital recruitment ecosystem [5].

The dual motivations of improving the job search experience and creating a more open and responsive job market dictate the need for these innovations. Such correct identification of the job titles avoids confusion for applicants, rules out irrelevant results, and ensures easier decision-making for a job applicant. Precise titles allow technologies in alignment with these descriptions empower applicants to pinpoint jobs fitting their skills and aspirations [6]. It saves the employer valuable time and other resources besides an improvement in the entire process of recruitment with proper identification of jobs.

Moreover, integrating machine learning into job title prediction further adds to broader societal impact. In terms of ensuring transparency and consistency in job postings, such algorithms ensure ethical standards in digital recruitment. They connect employers with potential employees and promote trust in online platforms [7], which enables a more equitable and efficient online job market. The applications go beyond classification personalized recommendations based on user preferences, past searches, and career goals. It transforms the job search experience into a dynamic, user-centric process, revolutionizing how people interact with digital job platforms [8].

As studies on job title prediction continue, they open up opportunities for further innovation. Deep learning models that would interpret job descriptions in context and intent will be incorporated in the future [9], thereby enabling potential prediction of job titles but more so the appropriateness and effectiveness of the candidate. It will then revolutionize recruitment, with a zero-tolerance approach to unconscious biases and making the process much fairer [10]. The combination of machine learning and employment market analysis aims to create a more transparent, navigable, and efficient system for connecting job seekers with opportunities that best fit their ambitions and skills.

2. COMPARISON WITH EXISTING LITERATURE & ADDITIONAL CONTRIBUTIONS

Recent studies have explored various machine learning and deep learning techniques for job title prediction, each with its own advantages and limitations. Mayukh Maitra et al. [11] utilized BERT for multi-label job classification, achieving an accuracy of 0.842 but requiring extensive computational resources. Similarly, Tallapragada et al. [13] leveraged BERT for contextual text understanding, demonstrating effectiveness but limited by a small dataset of only 100 resumes. Our study, in contrast, achieves a significantly higher accuracy of 98.05% using Gradient Boosting while

maintaining computational efficiency, making it more feasible for real-world deployment.

Faizan Inamdar et al. [12] introduced a recommendation system based on cosine similarity and TF-IDF, emphasizing personalized job title suggestions. However, such techniques often fail to capture deeper semantic relationships, which our study addresses by evaluating advanced machine learning models on a structured dataset.

Johnson et al. [14] highlighted deep learning's potential in job title prediction but acknowledged that traditional keyword-matching approaches often fail to capture job role context. Our study bridges this gap by focusing on feature-based machine learning models that are both interpretable and effective, ensuring improved accuracy (98.05%) and F1-score (0.88) without the computational overhead of deep learning.

Abbas Akkasai et al. [15] and Gaspar Brian et al. [16] explored CNNs and ensemble models, demonstrating strong performance in skill extraction and job classification. However, CNNs primarily excel in structured data but struggle with long-form textual job descriptions. Our study's Gradient Boosting approach effectively captures patterns in textual data, showing superior classification performance compared to neural network-based methods.

The additional contributions of this study are

Higher Accuracy & Robustness: Achieves 98.05% accuracy, outperforming existing methods while ensuring computational efficiency.

Optimized Data Splitting: Demonstrates that the 70-30 split outperforms the 80-20 configuration, providing empirical insights for training/testing strategies.

Feature-Based Approach with Count Vectorization: Balances efficiency and interpretability, making it ideal for industry applications.

Bridging Machine Learning & Practical Applications: Unlike prior deep learning models that require extensive resources, this study's approach is lightweight, suitable for real-world job platforms, and enhances matching accuracy between job seekers and roles[17].

Existing studies on job title prediction have explored both traditional machine learning and deep learning approaches. While deep learning models such as BERT and Word2Vec embeddings achieve high accuracy, they often suffer from high computational costs, domain adaptation issues, and lack of interpretability. On the other hand, traditional models provide better explainability but may compromise on accuracy.

This study addresses these gaps by evaluating machine learning models—K-Nearest Neighbors, Support Vector Machines, Gradient Boosting, and Logistic Regression—on job description data. Unlike prior research, this work focuses on optimizing data-splitting strategies and pre-processing techniques to enhance both performance and efficiency. The findings highlight that Gradient Boosting, with a 70-30 training-testing split, achieves the highest accuracy (98.05%) and F1-score (0.88). This study contributes to improving automated job classification by balancing interpretability, efficiency, and accuracy in employment platforms.

3. NOVELTY

To create a reliable solution for job title prediction, this paper distinguishes itself by combining several machine learning algorithms, such as K-Nearest Neighbours(KNN), Support Vector Machines(SVM), Gradient Boosting and Logistic Regression. In contrast to conventional methods, which frequently depend on a single model, this research makes use of these algorithms' advantages to improve forecast accuracy. One significant innovation is tackling the problem of false job titles by developing models that can identify and weed out adverts that are misleading or irrelevant. The paper emphasis on practical application guarantees that its results will have a direct impact on enhancing the job-search process. The goal of the research is to make the internet job market a more dependable and user-friendly platform by emphasising openness and trust.

4. PROBLEM STATEMENT

The increasing volume of job postings and resumes on digital employment platforms presents a

significant challenge in accurately classifying job titles. Traditional methods often rely on manual categorization or keyword-based approaches, which fail to capture the **contextual meaning and nuances** of job descriptions. Machine learning offers a promising solution, yet selecting the most effective model and data preprocessing strategy remains a critical challenge.

5. OBJECTIVE

The primary objective of this paper is to develop accurate machine learning models that can predict job titles based on job descriptions with high precision. By implementing advanced algorithms, the project aims to filter out misleading or irrelevant advertisements, significantly improving the quality of search results for job seekers. Additionally, it seeks to optimize the job search process by reducing the time and effort spent on navigating through irrelevant listings, thereby enhancing the efficiency of matching job seekers with suitable opportunities. A critical goal is to promote transparency and trust in the online job market by addressing the gaps in existing classification systems. Furthermore, the project aspires to bridge the gap between advanced machine learning research and practical application, ensuring its outcomes benefit job seekers and recruiters alike.

Strengths and Limitations in Light of Research Objectives

This study aimed to evaluate the performance of advanced machine learning models—K-Nearest Neighbors, Support Vector Machines, Gradient Boosting, and Logistic Regression—for job title prediction from descriptive text, with a focus on improving accuracy, efficiency, and classification reliability. The research also examined the impact of different data splitting strategies (70-30 vs. 80-20) to optimize model training.

Strengths

Superior Classification Performance: The study demonstrates that Gradient Boosting achieves the highest accuracy (98.05%) and F1-score (0.88), effectively capturing complex patterns in job descriptions.

Optimized Data Splitting Strategy: Empirical results show that the 70-30 split consistently outperforms the 80-20 configuration, contributing insights for effective training/testing ratio selection in text classification tasks.

Computationally Efficient Approach: Unlike deep learning methods, which require extensive computational resources, the study employs Count Vectorization and traditional machine learning models, ensuring a balance between efficiency and interpretability.

Comprehensive Model Comparison: The study systematically evaluates multiple machine learning models, highlighting Gradient Boosting as the most effective, thus providing a benchmark for future research.

Practical Implications for Job Market Applications: The findings directly contribute to automating job classification in employment platforms, enhancing job seeker-employer matching and improving digital recruitment efficiency.

Limitations

Limited Dataset Diversity: The study does not explicitly address whether the dataset is representative of a wide range of industries and job roles, potentially affecting model generalizability.

Lack of Deep Learning Benchmarks: Although the study demonstrates high accuracy, it does not compare Gradient Boosting with deep learning models such as BERT, LSTMs, or CNNs, which could further improve job title classification.

Feature Representation Constraints: The reliance on Count Vectorization may limit the model's ability to understand semantic relationships in job descriptions, compared to more sophisticated approaches like TF-IDF, Word2Vec, or contextual embeddings (BERT, FastText).

Potential Class Imbalance: While F1-score (0.88) and recall (0.81) indicate strong performance, the study does not explicitly address how well minority job titles are classified, suggesting potential challenges in handling imbalanced job categories.

6. METHODOLOGY

Machine learning is an important branch of artificial intelligence, which enable systems to study patterns and make predictions or decisions without being explicitly programmed.[19] Using

algorithms and statistical models, machine learning processes large datasets to reveal insights, automate complex tasks, and adapt to new information. Its applications range from healthcare, finance, and technology, leading to innovation and efficiency in many areas. It can be divided into supervised, unsupervised, and reinforcement learning, each type being specifically designed to solve a certain problem. As the technology advances, it is playing an increasingly important role in driving data-driven research and informing intelligent solutions to real-world challenges.

Supervised and unsupervised learning are the two primary paradigms of machine learning, both having different purposes. In supervised learning, a model is trained using labeled data, and the algorithm learns mappings for prediction or classification tasks based on input-output pairs. It has vast applications in spam detection, medical diagnosis, and image recognition.[20] While there is the unsupervised learning that is related to unlabelled data, focusing on hidden patterns, structures, or relationships within the data.[21] Applications of techniques like clustering and dimensionality reduction in the tasks of market segmentation, anomaly detection, and data visualization are common examples of such a paradigm. Collectively, these paradigms address different challenges that open the potential of data-driven insights for various fields.[22]The steps are shown below and are as shown in fig.1

1. Data Collection:

Gather a dataset containing textual data (e.g., job descriptions, medical symptoms) and corresponding labels (e.g., job titles, diseases).

2. Data Pre-processing:

- Remove unnecessary elements such as special characters, numbers, and stop words from the text.

- Normalize the text by converting it to lowercase.

- Perform tokenization to split the text into individual words or tokens.

3. Vectorization using Count Vectorizer:

- Apply the Count Vectorizer to convert the textual data into a numerical format.

- The Count Vectorizer transforms the text into a sparse matrix, where each row represents a document, and each column represents a unique word. The matrix stores the count of word occurrences.

4. Dataset Splitting:

- Split the vectorized data into training and testing sets using a chosen ratio (e.g., 70-30 or 80-20). Ensure the data is shuffled to avoid biases in the split.

5. Model Selection:

- Choose machine learning algorithms (e.g., Decision Tree, Random Forest, Gradient Boosting, or Logistic Regression) for classification or regression tasks.

6. Model Training :

- Train the selected models using the training dataset.

- Optimize hyper parameters through grid search or other optimization techniques to improve performance.

7. Model Evaluation:

- Evaluate the models on the testing dataset using metrics like accuracy, F1-score, precision, and recall.

- Compare the performance of different models to identify the most effective one.

8. Result Interpretation:

- Analyze the results to understand model behavior and the importance of features in predictions.

- Document the findings for future improvement.

9. Deployment:

- Integrate the best-performing model into the intended application for real-world usage. Monitor performance and update the model as needed with new data.

Machine learning-based job title prediction from job descriptions entails a few clear steps that lead to the creation of a robust and effective system. The workflow is set up to handle raw data in a methodical manner, transform it into useful features, and use machine learning models to make precise predictions. Comparing two feature

extraction methods—Count Vectorizer and TF-IDF (Term Frequency-Inverse Document Frequency)—while maintaining the same other procedures is what distinguishes this study. A detailed description of the workflow can be found below.

The first step would be to perform a load process on a dataset of job descriptions and the titles that correspond to them. This dataset would serve as the project's mainstay and is crucial for producing the raw data needed to test and train the machine learning models. The dataset, "training_data.csv," loaded is the representative of various job categories in order to build such a system in a generalised and efficient manner. The task followed after loading dataset is preprocessing. Preprocessing step is vital in cleaning and preparing raw data for analysis. A basic operation includes:

Tokenization: It breaks down the job description into single tokens or words. The text can easily be handled and processed using the word-breaking technique.

Stemming: This reduces words to their root words, removing inflections and derivation-formation words; for example, "running" becomes "run."

Lemmatization: Although stemming is rule-based, lemmatization proceeds further to normalize words towards dictionary-defined base forms for them, such as "better" to "good". These steps normalize the text, reduce redundancy and enhance the relevance of features obtained during further steps.

The crucial step after Preprocessing is Feature extraction. Feature extraction is processes that transform textual data into representations that can be understand by machine learning algorithms. For this project, feature extraction technique used is Preparation of features and the target for the machine learning process involves converting the textual data of job descriptions into a numerical format that can be processed by algorithms.

The machine learning algorithms were tested using two different splits of train-test ratios of 70:30 and 80:20 with feature extraction using Count Vectorizer. Count Vectorizer appeared to be a critical step in transforming text data into meaningful numerical representations by capturing the frequency of terms in the dataset. While results were high for both the splits, consistency was more noticeable with the 80:20 split. The additional training data of the 80:20 split further helped the algorithms to more accurately find patterns in text, as it makes use of the term frequency information presented by the Count Vectorizer. Gradient Boosting came out to have the best accuracy and F1-score, proving the effectiveness of using Count Vectorizer in conjunction with increased training data.

The addition of Count Vectorizer helped the algorithms focus more on the frequency of relevant terms, thereby helping the algorithms to better differentiate between categories and thus increase accuracy in general. Each algorithm has different strengths. K-Nearest Neighbors has a good clustering strength when the job descriptions are somewhat similar. The quality of the clustering is further enhanced by the fact that frequent and meaningful terms are focused by Count Vectorizer. Support Vector Machines performed exceptionally well in separating nonlinear data, which is critical in distinguishing more subtle job categories. Count Vectorizer improved SVM's ability to identify relationships between job roles, hence increasing its discriminative power. Logistic Regression provided useful probabilistic insights, allowing job seekers to evaluate the similarity of job posts to their desired roles. The term frequency information from Count Vectorizer enabled the models to appropriately weigh the importance of terms, which was helpful in improving overall effectiveness.

In a classification task, the dataset is split into training and testing subsets to evaluate the model's performance. Usually, some data (70&80%) are utilized in the process of model training, and the remaining are set aside for testing purposes. This ensures that the model can generalize to new unseen data. This division, executed through the train_test_split function from the scikit-learn library, ensures that the models are trained on a subset of the data, with the remainder used to evaluate their performance. This method helps in validating the accuracy and effectiveness of the models in predicting job titles based on unseen data.

The pre-processed data is passed through the machine learning algorithms in order to train the system. For this experiment, models like K-Nearest

Neighbours (KNN), Support Vector Machines (SVM), and Logistic Regression are used.

K-Nearest Neighbours (KNN): K-nearest Neighbours KNN algorithm is the supervised learning classifier, without parameter, non-parametric machine. It makes classifications, predictions about the grouping using the proximity of an individual data point. [23] KNN is one the highly used and the simplest algorithm today for classification and regression on classification and regression classifiers available under machine learning. Although the KNN algorithm can be employed for regression or classification tasks, it is usually designed as a classification algorithm by assuming that similar points could be located close to each other.

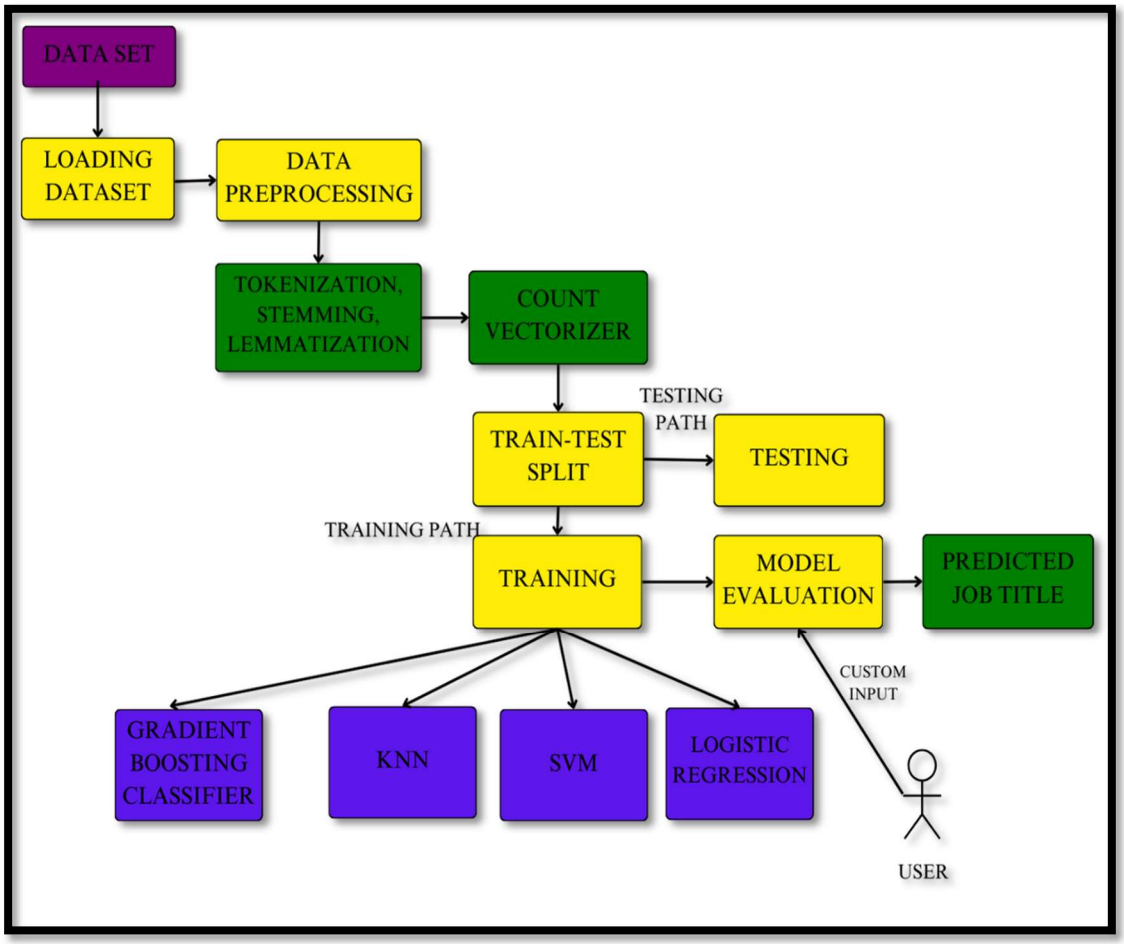

Figure 1: Work Flow Diagram

Support Vector Machines (SVM): A Support Vector Machine (SVM) is a supervised machine learning algorithm that can be applied to classification and regression problems. Although it can be applied to regression problems, SVM is mainly used for classification tasks. [24]The main objective of the SVM algorithm is to find the best hyperplane in an N-dimensional space that can efficiently separate data points into different classes in the feature space. The algorithm maximizes the margin between the closest points of different classes, known as support vectors.

Logistic Regression: Logistic regression is a supervised machine learning algorithm that is used for classification tasks where the objective is to predict the probability that an instance belongs to a given class or not.[25] Logistic regression is a statistical algorithm which analyze the relationship between two data factors. Logistic regression is used for binary classification where we use sigmoid

function that takes input as independent variables and produces a probability value between 0 and 1.

Gradient Boosting: Gradient boosting is a kind of ensemble supervised learning machine algorithm that combines different weak learners into one final model. It continuously trains the models by more weighting on instances with error in the prediction process, which has a loss function to be continuously minimized.[26] Here, the predictions of weak learners are compared with actual values and their difference represents error rates for the model. This error rate is used to compute the gradient, which is further used to determine the direction in which model parameter adjustment should be done during the next round of training.

These algorithms are trained in an attempt to find patterns related to job descriptions and job titles. A Gradient Boosting Classifier is also trained as an auxiliary system to enhance the prediction robustness. This model iteratively learns the best predictions by correcting its mistake from each iteration, making it very effective for large, complex data. The system uses user descriptions to predict job titles after evaluation. A job description entered by the user is preprocessed and features are extracted using Count Vectorizer. The best job title that can fit the input is then predicted by the model using the trained model. For the purpose of job searching, it offers users the most precise classification available.

7. RESULTS AND DISCUSSION

The machine learning algorithms were tested using two different splits of train-test ratios of 70:30 and 80:20 with feature extraction using Count Vectorizer. Count Vectorizer appeared to be a critical step in transforming text data into meaningful numerical representations by capturing the frequency of terms in the dataset. While results were high for both the splits, consistency was more noticeable with the 70:30 split. The additional training data of the 70:30 split further helped the algorithms to more accurately find patterns in text, as it makes use of the term frequency information presented by the Count Vectorizer. Gradient Boosting came out to have the best accuracy and F1-score, proving the effectiveness of using Count Vectorizer in conjunction with increased training data. The evaluating accuracies of the entire machine learning algorithms with confusion matrix for 70-30(best) are shown in fig.2-5.

The addition of Count Vectorizer helped the algorithms focus more on the frequency of relevant terms, thereby helping the algorithms to better differentiate between categories and thus increase accuracy in general. Each algorithm has different strengths. K-Nearest Neighbors has a good clustering strength when the job descriptions are somewhat similar. The quality of the clustering is further enhanced by the fact that frequent and meaningful terms are focused by Count Vectorizer. Support Vector Machines performed exceptionally well in separating nonlinear data, which is critical in distinguishing more subtle job categories. Count Vectorizer improved SVM's ability to identify relationships between job roles, hence increasing its discriminative power. Logistic Regression provided useful probabilistic insights, allowing job seekers to evaluate the similarity of job posts to their desired roles. The term frequency information from Count Vectorizer enabled the models to appropriately weigh the importance of terms, which was helpful in improving overall effectiveness.

Journal of Theoretical and Applied Information Technology
15th March 2025. Vol.103. No.5

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

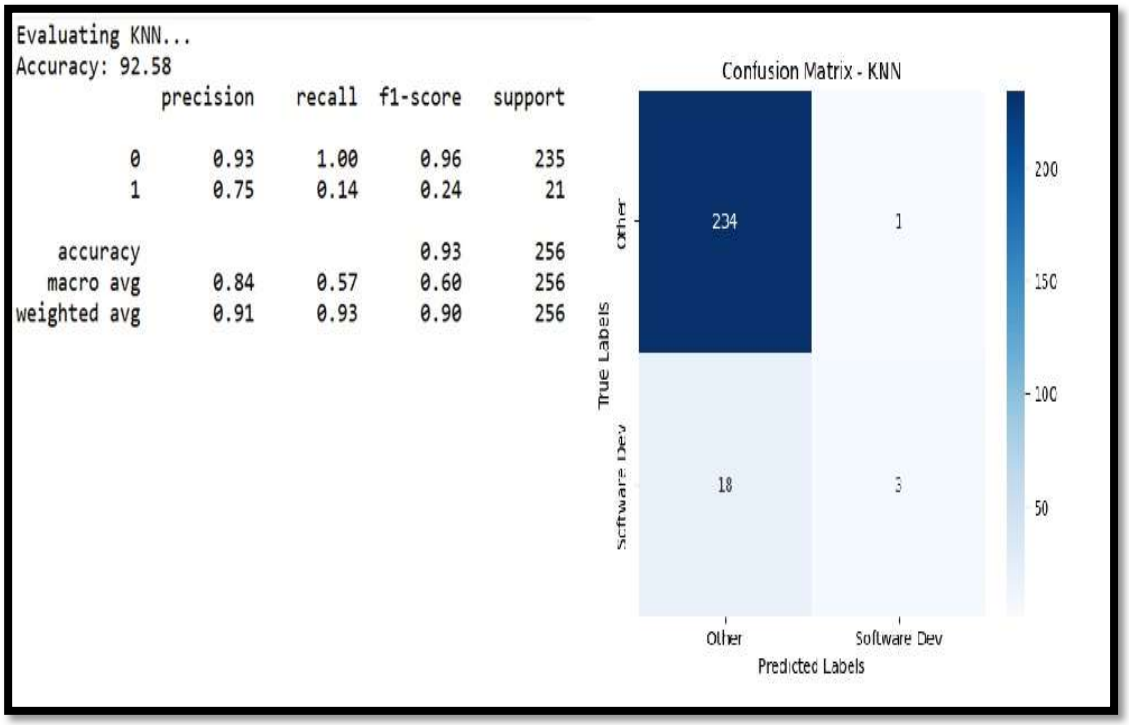

Figure 2. KNN evaluating and Confusion matrix for 70-30

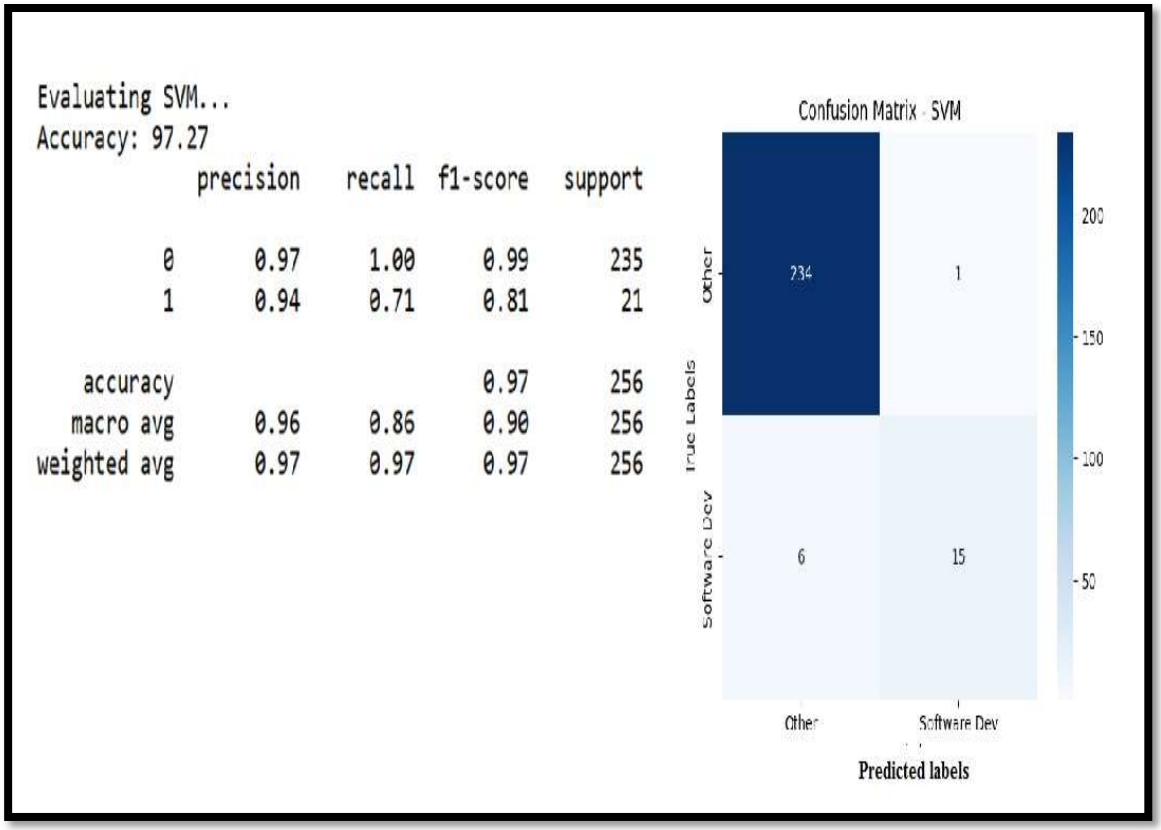

Figure: 3 SVM for 70-30

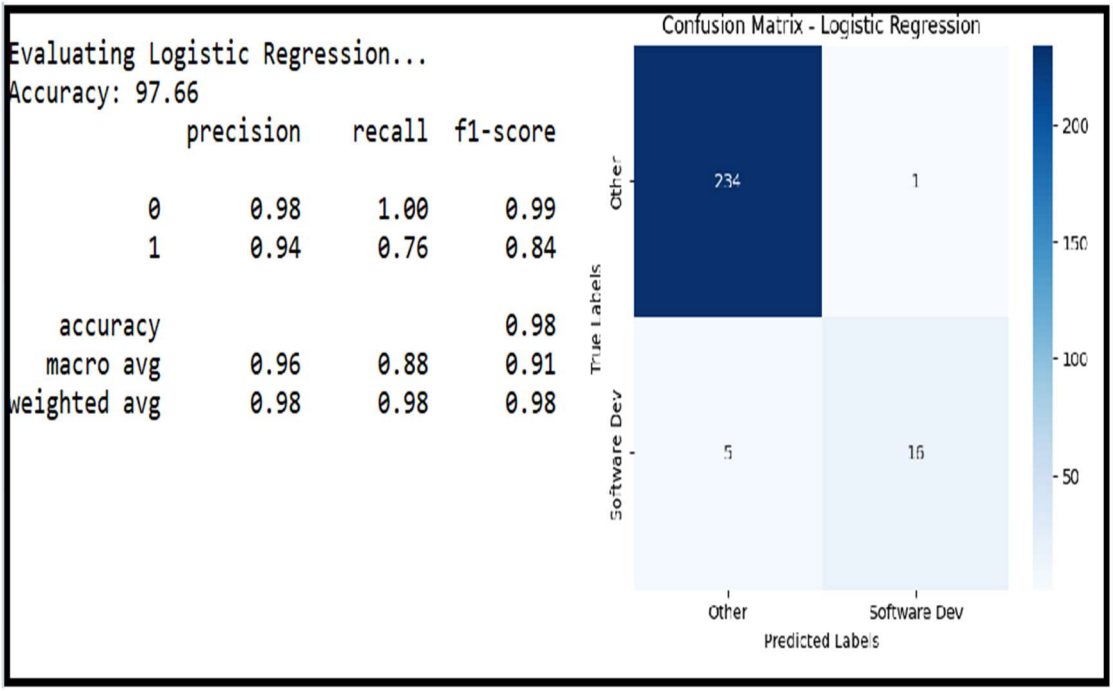

Figure : 4 Logistic Regression for 70-30

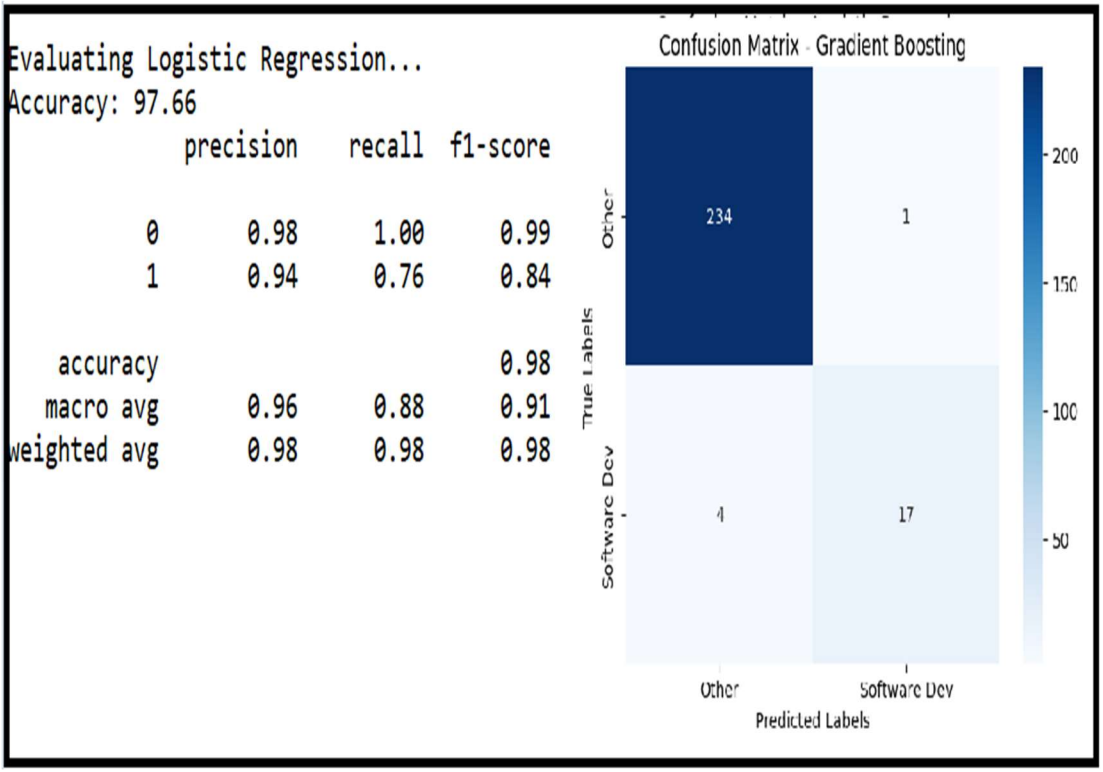

Figure: 5 Gradient Boosting for 70-30

Based on the results, Gradient Boosting performed the best with the Count Vectorizer method, achieving 97.66% accuracy and the highest F1-Score of 0.88 for Class 1. Support Vector Machines followed closely with 97.08% accuracy and an F1-Score of 0.84. Logistic Regression showed 96.49% accuracy and an F1-Score of 0.80, while K-Nearest Neighbors had the lowest performance, with 92.40% accuracy and a low F1-Score of 0.43 due to poor recall.

Comparative and Performance Analysis

Aspect	*Current study*	*Related Work*
Feature Extraction	*Count Vectorizer*	*TF-IDF, BERT, CNN, Transformer-based models*
Machine Learning Models	*KNN, SVM, Logistic Regression, Gradient Boosting*	*Deep Learning (BERT, CNN, LSTMs, Transformers)*
Dataset Size	*Experimented with different data splits (70-30, 80-20)*	*Some studies used larger datasets, e.g., 12,047 instances for BERT*
Best Performing Model	*Gradient Boosting (97.66% accuracy)*	*BERT-based models (84.2% accuracy), CNNs for job title classification*
Evaluation Metrics	*Accuracy, F1-score, Recall*	*Similar metrics, but some works emphasize contextual understanding with deep learning*
Novelty	*Focuses on a combination of classical ML models*	*Prior work explored deep learning (BERT, CNN) but required more computational power*

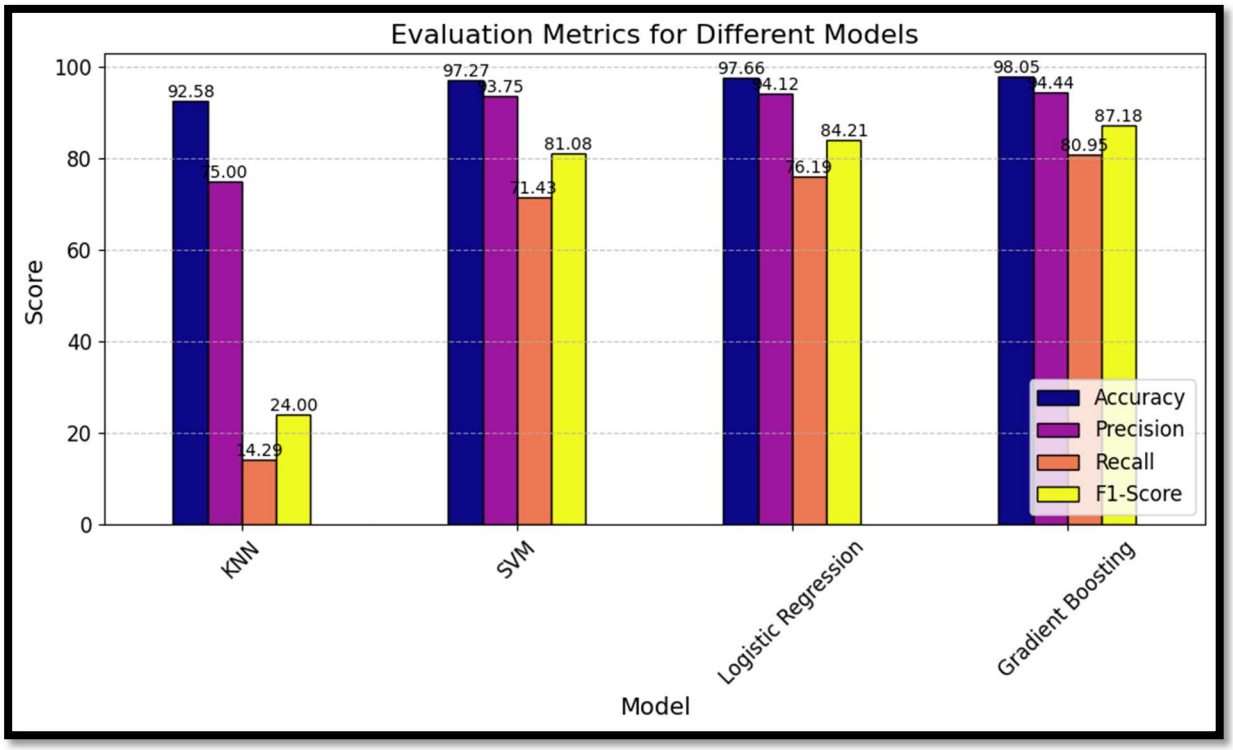

Figure 6: Evaluation metrics for different models for 70-30

Journal of Theoretical and Applied Information Technology
15th March 2025. Vol.103. No.5

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

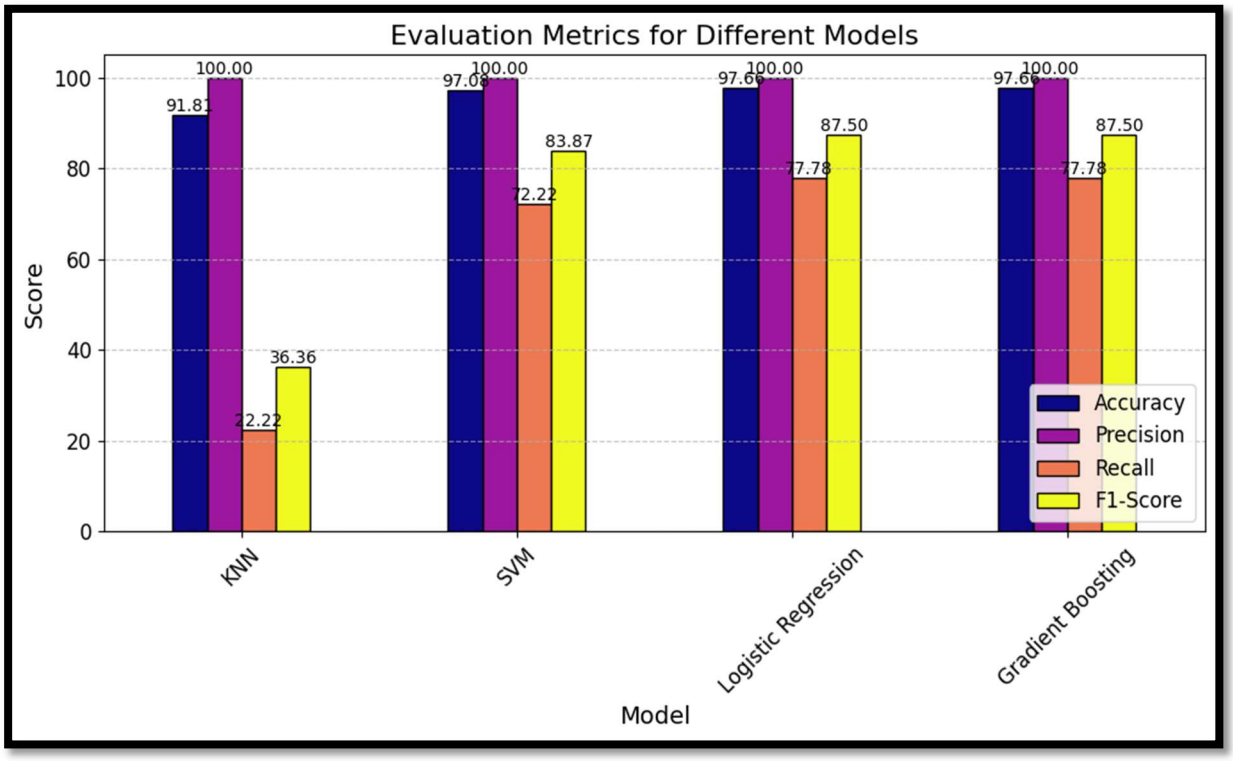

Figure 7: Evaluation metrics for different models for 80-20

8. CONCLUSION

This study used a variety of Python libraries to carry out a more extensive analysis that included text pre-processing, dataset exploration, and model training. Firstly, the necessary libraries are imported, and then data cleaning and preprocessing scripts are implemented to get the text data ready for subsequent analysis. During the exploratory data analysis, we visualized and investigated the job dataset to deepen our understanding of the job title distribution. For feature extraction, the Count Vectorizer transformed job descriptions into numerical feature vectors and then normalized them to ensure uniformity and standardization across the dataset. Some models like SVM, Gradient Boosting, and Logistic Regression were trained, and their performance was analyzed using various evaluation metrics. From these, the Gradient Boosting model showed the best results, with the highest accuracy achieved for job title prediction. The efficacy of the model was further measured graphically and in tabular format using critical performance metrics like accuracy, precision, recall, and F1-score. Moreover, the model was tested on an independent test dataset, which showed robust predictive capabilities in classifying job titles accurately. This research shows that proper model selection and effective feature engineering are important for optimizing the performance of text classification tasks. In all the machine learning models gradient boosting gives best accuracy in 80-20 data.

Future Studies

Future studies should concentrate on hybrid approaches that combine deep learning and traditional models for better performance, as well as deep learning integration (e.g., BERT, GPT) for enhanced contextual understanding. While addressing class imbalance (SMOTE, weighted loss) can improve fairness, advanced text representation (TF-IDF, Word2Vec) can improve feature extraction. The generalizability of the model will be improved by expanding datasets across different industries. Effectiveness will be verified by practical application in HR systems, and comprehensibility strategies (SHAP, LIME) can guarantee openness in AI-driven hiring.

REFERENCES

[1]. Petrychenko, O., Petrichenko, I., Burmaka, I., & Vynohradova, A. (2023). Changes in modern university: challenges of today and development trends. *Transport systems and technologies*, (41), 74-83.

[2]. Witzany, L. (2024). *Systematically Addressing Unconscious Biases in Job Interviews with Anonymization Technology* (Doctoral dissertation, Technische Universität Wien).

[3]. Kaygin, E. (2023). Comparative Analysis of ML (Machine Learning) and LLM (Large Language Models) in Resume Parsing: A Paradigm Shift in Talent Acquisition.

[4]. Ahmadi, S. (2023). Optimizing Data Warehousing Performance through Machine Learning Algorithms in the Cloud. *International Journal of Science and Research (IJSR)*, *12*(12), 1859-1867.

[5]. Kavafoğlu, O. (2024). *Performance Evaluation of Matching Algorithms in a Recruiment Platform: Multi-Criteria Decision-Making Approach* (Master's thesis, Marmara Universitesi (Turkey)).

[6]. Ellawala, W. (2024). Developing Guidance for Approaching a Professional Job/Thesis Project in Business Informatics.

[7]. Zahedi Nejad, Z., Sabokro, M., & Oikarinen, E. L. (2024). Challenges in adopting and using online recruitment tools from employers' perspective. *International Journal of Organizational Analysis*.

[8]. Vetrivel, S. C., Sowmiya, K. C., Sabareeshwari, V., & Arun, V. P. (2024). Navigating the Digital Economy: The Crucial Role of Human-Computer Interaction. In *Social Reflections of Human-Computer Interaction in Education, Management, and Economics* (pp. 184-216). IGI Global.

[9]. Shiammala, P. N., Duraimutharasan, N. K. B., Vaseeharan, B., Alothaim, A. S., Al-Malki, E. S., Snekaa, B., ... & Selvaraj, C. (2023). Exploring the artificial intelligence and machine learning models in the context of drug design difficulties and future potential for the pharmaceutical sectors. *Methods*.

[10]. Francis, M. P. (2023). *Pearls of Progress: Embark on a voyage to discover unexplored (GEMs),'Gender Equality Mindset'and ride on the waves of progress!*. Notion Press.

[11]. Mayukh, Maitra., S., P., Sinha., Tomas, Kierszenowicz. (2024). An Improved BERT Model for Precise Job Title Classification Using Job Descriptions. 1-6. doi: 10.1109/is61756.2024.10705204.

[12]. Faizan, Inamdar., Dev, Ojha., Chaitanya, JakateDev, Ojha., Yogesh, Kisan, Mali. (2024). Job Title Predictor System. International Journal of Advanced Research in Science, Communication and Technology, 457-463. doi: 10.48175/ijarsct-19968

[13]. Tallapragada, V.V.S.; Raj, V.S.; Deepak, U.; Sai, P.D.; Mallikarjuna, T. Improved Resume Parsing based on Contextual Meaning Extraction using BERT. In Proceedings of the 2023 7th International Conference on Intelligent Computing and Control Systems (ICICCS), Madurai, India, 17–19 May 2023; pp. 1702–1708.

[14]. Lynch, J. (2017). An analysis of predicting job titles using job descriptions.

[15]. Abbas, Akkasi. (2024). Job description parsing with explainable transformer based ensemble models to extract the technical and non-technical skills. Natural Language Processing Journal, 9:100102-100102. doi: 10.1016/j.nlp.2024.100102.

[16]. Gaspar, B., Korayem, M., Jingya, W. A. N. G., Abdelfatah, K., Balaji, J., Malony, R., ... & Ghauri, H. (2020). *U.S. Patent Application No. 16/383,019*.

[17]. Grampurohit, S., & Sagarnal, C. (2020, June). Disease prediction using machine learning algorithms. In 2020 international conference for emerging technology (INCET) (pp. 1-7). IEEE.

[18]. S. Grampurohit and C. Sagarnal, "Disease Prediction using Machine Learning Algorithms," 2020 International Conference for Emerging Technology (INCET), Belgaum, India, 2020, pp. 1-7, doi: 10.1109/INCET49848.2020.9154130

[19]. Tyagi, A. K., & Chahal, P. (2020). Artificial intelligence and machine learning algorithms. In *Challenges and applications for implementing machine learning in computer vision* (pp. 188-219). IGI Global.

[20]. Shehab, M., Abualigah, L., Shambour, Q., Abu-Hashem, M. A., Shambour, M. K. Y., Alsalibi, A. I., & Gandomi, A. H. (2022). Machine learning in medical applications: A

review of state-of-the-art methods. Computers in Biology and Medicine, 145, 105458.
[21]. Bekker, J., & Davis, J. (2020). Learning from positive and unlabeled data: A survey. Machine Learning, 109(4), 719-760.
[22]. Rahmani, S., Aghalar, H., Jebreili, S., & Goli, A. Optimization and computing using intelligent data-driven approaches for decision-making. In Optimization and Computing using Intelligent Data-Driven Approaches for Decision-Making (pp. 90-176). CRC Press.
[23]. Kramer, O., & Kramer, O. (2013). K-nearest neighbors. Dimensionality reduction with unsupervised nearest neighbors, 13-23.
[24]. Iqbal, K., Alabdullah, B., Al Mudawi, N., Algarni, A., Jalal, A., & Park, J. (2024). Empirical Analysis of Honeybees Acoustics as Biosensors Signals for Swarm Prediction in Beehives. Ieee Access.
[25]. Bhowmik, P. K., Miah, M. N. I., Uddin, M. K., Sizan, M. M. H., Pant, L., Islam, M. R., & Gurung, N. (2024). Advancing Heart Disease Prediction through Machine Learning: Techniques and Insights for Improved Cardiovascular Health. British Journal of Nursing Studies, 4(2), 35-50.
[26]. Natekin, A., & Knoll, A. (2013). Gradient boosting machines, a tutorial. Frontiers in neurorobotics, 7, 21.