Machine Learning for Recruitment: Analyzing Job-Matching Algorithms

Aram Khasro Jirjees, Aram M. Ahmed, Abdulhady Abas Abdulla, Joan Lu, Ekhlas Mohammed Noori, Rozhgar Nasradeen Kareem, Bryar A. Hassan, Hadi Veisi, Tarik A. Rashid

Abstract

The limitations of traditional keyword-matching algorithms in capturing nuanced meaning and context pose significant challenges for job seekers and recruiters alike. These methods often fail to account for variations in how job descriptions and candidate profiles are phrased, leading to suboptimal matching. This paper investigates a BERT-based approach to improve job-applicant matching by leveraging deep contextual embeddings. Models such as SVM, LSTM, and MLP are trained using BERT embeddings derived from real-world job descriptions, and their performance is rigorously evaluated using skill datasets. The results highlight the superiority of BERT-enhanced models over standard keyword-matching techniques. Specifically, the SVM-based model achieved 94% accuracy, LSTM reached 92%, and MLP performed at 90%. These findings demonstrate BERT's capability to significantly improve the recruitment process by offering more accurate and context-aware job-candidate matches, ultimately enhancing both employer and job seeker experiences.

Keywords: Job Matching, BERT, Recruitment Systems, Deep Learning, SVM, LSTM, MLP, NLP

1. Introduction

In the rapidly evolving labor market, effectively aligning job seekers with suitable opportunities is essential. However, traditional job-matching methods, which heavily rely on keyword searches and manual processes, often prove inconsistent and inefficient. The primary shortcomings of these approaches stem from their inability to comprehend the semantic intricacies of job descriptions and resumes, delaying the identification of truly qualified candidates.

Traditional keyword-based recruitment methods struggle with issues such as polysemy (words with multiple meanings), synonymy (words with similar meanings), and an inability to grasp the context in which skills and qualifications are presented. As a result, employers may overlook well-suited candidates, while job seekers face difficulties in finding roles that align with their expertise. Earlier machine learning (ML) models also exhibited similar challenges, as they lacked the ability to fully understand semantic nuances, limiting their effectiveness in improving job-matching accuracy.

This study aims to explore the potential of machine learning, particularly deep learning-based natural language processing (NLP) models, in overcoming the shortcomings of traditional job-matching techniques (Awlla et al., 2025). Specifically, it investigates the transformative role of BERT (Bidirectional Encoder Representations from Transformers), a model introduced by Devlin et al. (2018), which significantly enhances NLP tasks by capturing contextual meanings in bidirectional training. The research focuses on leveraging BERT embeddings to improve the accuracy of job matching, bridging the gap between human language complexities and computational analysis.

To achieve its objectives, the suggested model proposes a hybrid methodology that integrates BERT with machine learning models, such as Support Vector Machines (SVM), Multi-Layer Perceptron (MLP), and Long Short-Term Memory (LSTM) networks. By transforming job descriptions and resumes into meaningful numerical representations, these algorithms can detect patterns and improve prediction accuracy (Abdullah et al., 2024). Additionally, previous research, such as Pendyala et al. (2022), has demonstrated how combining machine learning, web crawlers, and APIs can streamline recruitment processes, enhancing speed, accuracy, and reducing bias. This study builds upon such advancements to create a more efficient and adaptable job-matching system.

This research contributes to the ongoing development of AI-driven recruitment by evaluating the effectiveness of BERT-based embeddings in job matching. By addressing the limitations of keyword-based systems and earlier ML models, the study aims to provide a practical, scalable solution for improving hiring processes. The findings will offer insights into how BERT and machine learning can revolutionize talent acquisition, benefiting both job seekers and employers through enhanced efficiency, accuracy, and fairness in recruitment. Ultimately, this study paves the way for future innovations in AI-powered job matching, ensuring better alignment between skills and job opportunities in a dynamic labor market.

This paper comprises five sections covering theoretical and practical aspects. Section Two reviews research on HRM matching algorithms, BERT embeddings, and ML/NLP in job matching. Section Three details the methodology, including data preparation and evaluation metrics. Section Four highlights results, model comparisons, and discussions. Section Five concludes with key findings, significance, and future directions.

2. Related Work

AI is revolutionizing recruitment in HRM by enhancing job matching and recommendation systems. This study examines how advanced ML and NLP models improve accuracy and the efficiency of aligning candidates with the right roles. By reviewing key publications, it aims to provide deeper insights and contribute to the ongoing evolution of recruitment technologies. There have been significant efforts to create job recommendation systems. In some approaches the matching algorithms are improved, while in other studies the job matching and candidate assessment is also considered. This section reviews some of the most relevant and robust works in this area.

Beginning with Panchasara, Gupta, and Sharma 2023, who utilized web scraping and NLP technologies to gather and process data from 20 leading Indian IT job portals. A custom web scraper extracted relevant content, such as text, links, and images, for analysis. After preprocessing, a pre-trained BERT model was employed to create vector representations of users and job profiles, enabling accurate mapping between them. The integration of diverse data sources and advanced NLP algorithms led to robust data collection and representation strategies in building effective matching systems.

Mixed methods matching combines various techniques for better accuracy and robustness, blending rule-based methods with ML or deep learning models. (Homsapaya and Budsara 2020) combined ML, DataMart models, and ETL processes to streamline job alignment, tracking trends and employer-applicant interactions. Also, (Chou and Yu 2020) leveraged big data, ML, and text mining to match candidates with employers at job fairs.

Model comparison and optimization help identify the most suitable algorithms. For example, (Smelyakov, Hurova, and Osiievskyi 2023) focused on selecting the right ML model for business decisions by comparing the performance of different algorithms using collected data. The data was split into training and testing sets, and several benchmarks, including throughput time, accuracy, precision, recall, and F1 measure, were used to assess the algorithms' performance (Abdullah et al., 2025).

Moreover, Chowdhury, Chowdhury, and Sultana 2022 This proposed system was linked employment seekers' CVs to the jobs, increasing the efficiency and effectiveness of employment seekers. The system related job descriptions, duties, and requirements to candidates' experience, projects, and skills. It used word vectors from the GloVe (Global Vectors for Word Representation) model to calculate similarity, achieving an accuracy of 79.2%. The system utilized co-occurrence data, the frequency with which words appear together in a corpus, to generate word vectors and capture word meanings and associations. This allowed companies to quickly eliminate unsuitable candidates based on their scores.

Finally, (Deshmukh and Raut 2024) introduced a brand new concept of an automated resume filtering technique using the latest advancements in NLP including Google's BERT. Their method involved gathering job descriptions from Glassdoor and obtaining resumes from 200 participants with permission. They identified key behavioural competencies, searched for specific terms in the resumes, and prioritized the most relevant attributes. The preprocessing included removing punctuation, stop words, and identifying important keywords. Lemmatization and stemming were used for accuracy. Using a BERT model, they calculated a similarity index to match candidates, screening one resume per second with a similarity score of up to 0.3. BERT-based NLP techniques improved speed, accuracy, and efficiency, reducing hiring time and enhancing decision-making, demonstrating its potential to transform automated recruitment. Table 1 provides a summary of the literature review.

Table 1 : The summary of Literature review.

Authors	Types	Contribution	Limitation
(Panchasara, Gupta, and Sharma 2023)	Web scraping, NLP, BERT	Created a job suggestion system using data from an Indian IT job portal.	Focused on Indian IT sector; model may not be applicable to other industries or regions.
(Pendyala et al. 2022)	ML, NLP	Matched job seekers with positions based on keywords, skills, and education.	Relies on accurate self-reported data from job seekers; may not capture all relevant factors.
(Kaya and Bogers 2023)	BERT	Adapted BERT for matching resumes with job descriptions, showing improved performance compared to others.	Limited evaluation on a single dataset may not generalize to other domains or languages.
(Kurek et al. 2024)	all-MiniLM-L6-v2, Similarity	Enhanced matching accuracy using advanced pre-trained models and similarity metrics.	May require substantial amounts of data and computational resources for training and fine-tuning.
(Pang 2024)	Word2Vec, ML	Explored Word2Vec for vectorizing resumes and job descriptions for data engineering positions.	Limited to data engineering; may not generalize to other job types.
(Khan et al. 2023)	Machine Learning	Predicted job satisfaction and recommended suitable jobs using various ML techniques.	Relies on self-reported data on job satisfaction; may not capture all aspects of job satisfaction.
(Chen 2022)	GBT-CNN	Proposed a GBT-CNN algorithm to capture deeper correlations between job seeker and job information.	Algorithm complexity may hinder interpretability and explainability.
(Bhatia et al. 2019)	BERT	Built a resume parser and used BERT for ranking candidates based on suitability.	Requires significant computational resources for BERT-based classification.
(Alsaif et al. 2022)	Content-Based Filtering	Recommended jobs to job seekers based on resume skills using content-based filtering.	Limited to explicit skills mentioned in resumes; may not capture latent skills.
(Chowdhury, Chowdhury, and Sultana 2022)	NLP, GloVe	Used GloVe to match resumes with job postings, achieving 79.2% accuracy.	Relies on GloVe embeddings; may not capture all semantic nuances of resumes and job postings.
(Sudha et al. 2021)	Multiple-choice questions Logistic Regression	Developed an online application for candidate registration, personality analysis, and CV matching using Logistic Regression.	The specific details of the personality analysis and CV matching algorithms, and their potential limitations, are not explicitly mentioned
(Homsapaya and Budsara 2020)	ML, Data Mart Modeling	Combined ML and data mart modeling for job matching, considering skills and disease limitations.	Data mart modeling may be computationally expensive and complex to implement.
(Appadoo, Soonnoo, and Mungloo-Dilmohamud 2020)	ML, Recommender Systems	Developed "Job Fit," a job recommendation system using ML and recommender systems.	Model performance depends heavily on the quality and relevance of historical data.

Authors	Types	Contribution	Limitation
(Tejaswini et al. 2022)	Content-Based Filtering, KNN	Ranked CVs based on job descriptions using content-based filtering and KNN.	Relies on explicit content features; may not capture implicit relationships between jobs and CVs.
(Gadegaonkar et al. 2023)	Content-Based Filtering, ML	Developed an Android app for recommending IT jobs using content-based filtering.	Limited to the IT industry; model may not be adaptable to other sectors.
(Chou and Yu 2020)	Big Data, ML, Text Mining	Analyzed online discussions and recommended candidates for job openings at job fairs.	Requires substantial amounts of data for analysis; may not be feasible for smaller job fairs.
(Wang, Jiang, and Peng 2021)	BERT	Built a BERT-based model for person-to-job post matching.	Model complexity and computational requirements may hinder practical deployment.
(Green 2023)	NLP, BERT, ML	Explored NLP for entity recognition in job descriptions and used BERT for job application success prediction.	Limited by the quality and coverage of the entity recognition component.
(Schwartz et al. 2020)	BERT, Classifiers	Added classifiers to BERT layers for early stopping in text classification and natural language inference.	May not be suitable for tasks where high accuracy is critical at all costs.
(Bouhoun et al. 2023)	TSDAE, BERT	Combined TSDAE and BERT for specialized sentence embeddings in HR data.	Effectiveness of TSDAE may vary depending on the specific HR dataset used.
(Smelyakov, Hurova, and Osiievskyi 2023)	Various ML Algorithms	Evaluated multiple ML algorithms for hiring decisions.	Choice of evaluation metrics may influence the selection of the "best" algorithm.
(Rahmani, Groot, and Rahmani 2023)	CRISP-DM, Clustering, Classification	Used CRISP-DM with clustering and classification algorithms for job and labor market analysis.	Clustering and classification may oversimplify complex job and labor market dynamics.
(Honorati, Ferré, and Gajderowicz 2023)	N/A	Found that job matching tools need more information on job seekers' training, skills, and interests.	Highlights limitations of current job matching tools rather than solution.
(Ali et al. 2022)	NLP, ML	Used NLP and ML for resume classification, finding SVM to be the most accurate.	Focused on resume classification; may not address other aspects of job matching.
(Deshmukh and Raut 2024)	NLP, BERT	Used BERT for automated resume screening, showing promising results in speed and accuracy.	Relies on the quality of pre-trained BERT models; may not be adaptable to specific HR domains.

3. Methodology

This section offers a detailed review of the methods used to develop the improved matching system, including data collection, preprocessing, feature extraction, and machine learning models. It explains these methods to clarify the research process and the rationale behind each methodological choice. This study's methodology utilizes advanced machine learning techniques and BERT embeddings to address the task matching problem. It outlines the analysis methods, including data collection, preprocessing, feature engineering, model training, evaluation, and prediction of new job titles. The data, sourced from Kaggle, includes diverse job descriptions, skills, and titles. It undergoes careful preprocessing, where significant text fields are combined into a single composite field, forming the basis for feature extraction and ensuring its suitability for machine learning model training.

Feature engineering utilizes BERT to generate contextual embeddings that capture the nuances of job descriptions and requirements, providing robust features for machine learning models. LSTM networks, MLPs, and SVMs are trained and optimized using these embeddings, with their performance in job title classification evaluated through accuracy metrics. Last of all, the trained models are applied to other job descriptions to predict the most suitable job titles. This demonstrates the models' general applicability and their potential to enhance recruitment by providing more accurate matches between jobs and suitable candidates. The following sections detail each methodological aspect of the study, offering comprehensive insights into the data preparation, modeling, and evaluation processes that support this research. As shown in **Error! Reference source not found.**, that explains the study's approach.

Figure 1: Methodology Structure workflow for feature extraction and job matching predictions.

3.1 Data Collection and Preprocessing

The dataset for this study was obtained from Kaggle, a popular data science competition website with several data scientists and machine learning experts. Kaggle has a huge collection of data types that can be used for analytics and machine learning models. This study used the Synthetic Job Dataset, which is a comprehensive collection of artificially generated job advertisements with a structure and content similar to real-world job listings. This synthetic dataset's rich and diversified content makes it incredibly useful for data scientists, machine learning practitioners, and NLP researchers.

Data was collected in the form of a CSV file with 1,615,940 rows and twenty-three unique columns. Each row represents a unique job advertisement, and the dataset contains a variety of information such as job descriptions, skills, positions, and job titles. To make the data easier to examine, relevant text fields were combined into a single composite text field. This composite text box combines the job description and required abilities, capturing all relevant information about the job post in one location. The important elements addressed in this study are:

1. Job Description, define job tasks and requirements.

2. Skills, specify job-related skills and qualifications.

3. Job Title, indicate the advertised job title or role.

These features were selected for their importance in identifying the job and the necessary qualifications, making them essential for the job-matching process. The job title, which served as the goal variable, was encoded to train the ML model. That is the framework of data which is shown in Table 2 Framework of Data and column description.

Table 2: Framework of Data.

NO.	Column Name	Description
1	Job Id	Unique identifier for each job posting
2	Experience	Required experience for the job
3	Qualifications	Educational qualifications required
4	Salary Range	Salary range offered
5	Location	Job location
6	Country	Country of the job location
7	Latitude	Geographic latitude
8	Longitude	Geographic longitude
9	Work Type	Type of employment
10	Company Size	Size of the company
11	Job Posting Date	Date when the job was posted
12	Preference	Job preferences
13	Contact Person	Name of the contact person
14	Contact	Contact details
15	Job Title	Title of the job
16	Role	Role in the company
17	Job Portal	Job portal where it is posted
18	Job Description	Description of the job
19	Benefits	Benefits offered by the company
20	Skills	Required skills
21	Responsibilities	Job responsibilities
22	Company	Company name
23	Company Profile	Profile description of the company

Quick understanding of key requirements, the word cloud visually highlights capabilities, certifications, and highly pushed services, allowing for a quick understanding of key project concepts. Taken in the whole, using word cloud for skills provide a more effective way of presenting, describing, and improving the attention given to skills in different capacities of learners, educators, professionals, and people in general. Lastly, for recruiting the right set of talent, clear title increases visibility and enhances search capabilities. The general naming of jobs has a wide reach on the internet and since certain job positions have specific names, the positions will be well advertised and thus attract the right talent. As shown in the Figure 2

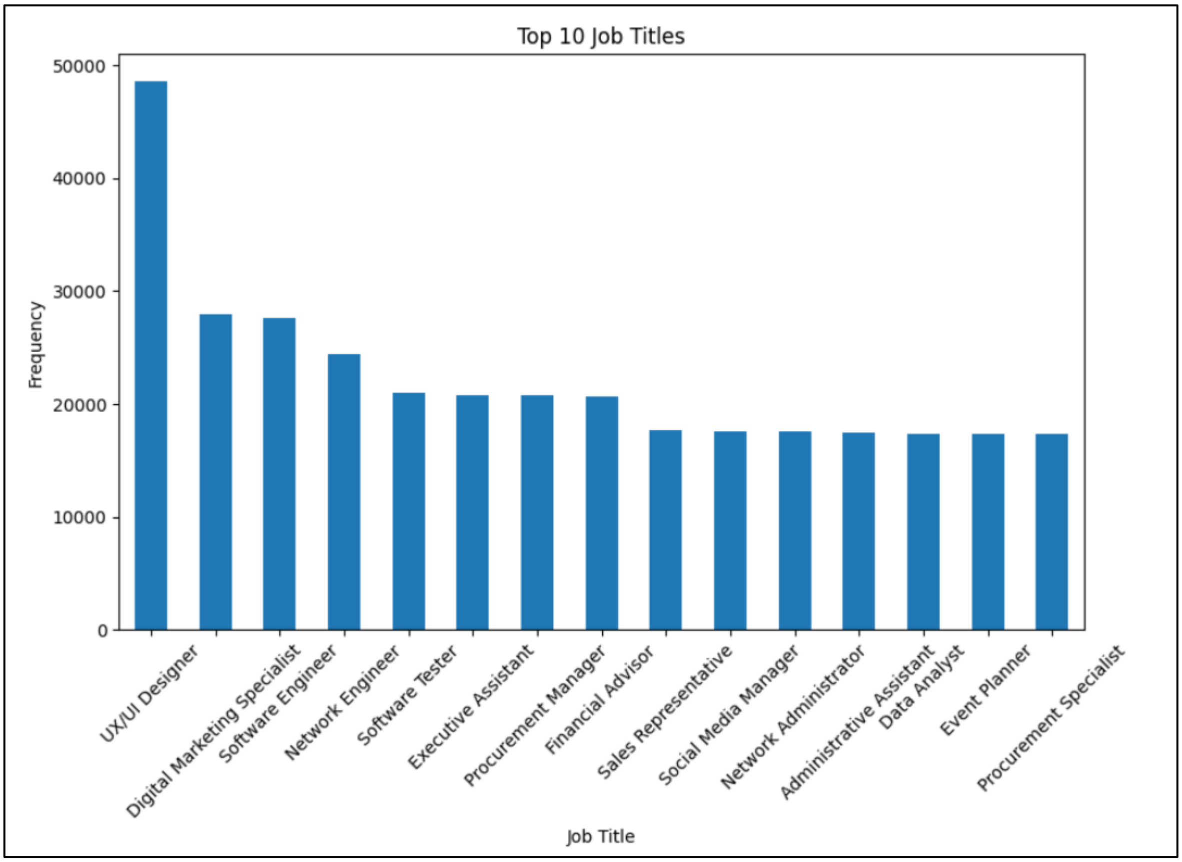

Figure 2: Top Job Titles in the data.

3.2 BERT for Feature Engineering

Feature engineering is a critical step in converting text data into numerical features that machine learning models can process. BERT, a powerful language model, is used for content extraction, transforming job descriptions and skills into mathematical representations (feature vectors) that capture the meaning of the text. The 'bert-base-uncased' model, consisting of 12 layers, 768 hidden states, and 12 attention heads, was employed for this task. As an 'uncased' model, it focuses on the semantic content without distinguishing between uppercase and lowercase letters. A tokenizer was utilized to ensure high precision in feeding text into the model. During feature extraction, input texts were tokenized, with padding and truncation applied to maintain a maximum sequence length of 512. These tokenized inputs were fed into BERT, which generated feature representations based on the [CLS] token, summarizing the key ideas of the job description and skills. Additionally, a Label Encoder was used to convert job titles into numerical labels for further processing in the machine learning pipeline.

By integrating BERT-derived features, our models gain access to rich, contextual aware representations of job descriptions and competencies, enhancing their ability to accurately predict job titles and match individuals to suitable opportunities. This contextual understanding is essential in recruitment, where words may have different meanings based on context.

3.3 Data Splitting and Training

In our work, to split the dataset into training and testing we used the function train_test_split from scikit-learn. We employed the following parameters:

1. features: This represents features extracted by BERT from job descriptions and skills, which serve as input data for our models.

2. Encoded_labels, these are the numerical encoded labels of job title as the target variable for prediction.

3. We split the data into test data and training data Test size equal to 0. 1 where the test data constituted 10% of the entire data while the training data constituted 90% of the whole data. Such a division allows us to have enough data for the learning of patterns and the evaluation of the model's ability to generalize.

4. The variable of Random state equals to 42, Which helps of random number generator, we secure randomness in splitting the data and the results of our experiment will be reproducible.

These data divisions facilitated efficient model training and evaluation, allowing most of the data to be used for training while reserving a portion for unbiased performance assessment.

3.4 Model training

Several machine learning models are trained based on the extracted BERT attributes. The models include BERT Classification, LSTMs, MLPs, and SVMs. Each of these models has advantages in processing different types of data.

3.4.1 Long Short-Term Memory

The model uses a type of RNN, specifically LSTM, to capture long-term dependencies in text sequences, making it ideal for classifying job titles and skills. The LSTM core layer, with 64 units, allows the model to learn and retain patterns across sequences, while the input shape is defined to handle BERT-derived features. The final dense layer has neurons equal to the number of job title classes, with a 'softmax' activation ensuring output values between 0 and 1, summing to 1. The Adam optimizer is used for efficient training, adjusting learning rates, and the model is trained for five epochs, processing 32 samples at a time. After training, the model's performance is evaluated on a test set to assess its generalization and avoid overfitting.

By employing BERT approach, we preprocess the data generated from job descriptions and skills and made it possible for the LSTM model to learn the context and dependencies to classify the right job title.

3.4.2 Multilayer Perceptron

The MLP model operates as a network of interconnected "neurons" organized into layers, processing data by passing information through these layers. With three hidden layers consisting of 150, 100, and 50 neurons, the model is designed to capture and learn complex patterns within the dataset. Each layer processes information from the previous one, enabling the model to develop a deep understanding of intricate relationships. The inclusion of 300 training iterations provides the model with ample opportunity to fine-tune its internal connections for improved pattern recognition.

Key parameters enhance the MLP's functionality. The ReLU activation function introduces non-linearity, allowing the model to handle complex relationships beyond the scope of linear models. The Adam optimizer adjusts model parameters during training to minimize errors and achieve optimal performance. Additionally, a fixed random state ensures reproducibility by initializing weights consistently across experiments. These features collectively make the MLP a powerful tool for recognizing and processing intricate data patterns. In job title classification, the MLP analyses BERT-derived features to identify patterns within job description and title data, uncovering hidden relationships for accurate predictions. After training, the model predicts job titles for the test set, which are then evaluated for performance.

3.4.3 Support Vector Machine

In this study, a Support Vector Machine (SVM) with a linear kernel was used for job title classification, leveraging its ability to find a hyperplane that maximizes the margin between classes. The linear kernel was chosen due to the high dimensionality of the BERT-generated features. The model was trained on the training dataset and then applied to the test set for job title prediction. The SVM's strength lies in its effectiveness at determining an optimal hyperplane for separating job title classes, providing a robust and interpretable method for classifying job titles based on the extracted textual features.

3.4.4 BERT Classification

To classify job titles across various job classes, we utilized a pre-trained BERT model designed for sequence classification tasks. This model, based on the 'bert-base-uncased' architecture, was fine-tuned to adapt to our dataset, with the output layer set to the number of job titles to predict. The Adam optimizer, specifically designed for transformer models, was used with a learning rate of 2e-5, typical in BERT fine-tuning. During training, the data was processed in batches, and weight adjustments were made based on the computed loss. The model's performance was then evaluated on a test set, which it had not encountered before, by comparing predicted labels with true labels.

3.5 Model Evaluation

Each model is evaluated based on performance metrics like accuracy, precision, and recall, which focus on minimizing false positives and false negatives. Accuracy provides an overall measure of correctness, while precision and recall specifically address errors related to positive and negative classifications. Additionally, confusion matrices are used to gain deeper insights into the models' predictive performance and identify their weaknesses.

3.6 Predicting New Job Descriptions

The primary objective of developing and evaluating the machine learning models in this work is to predict job titles for new job descriptions. In real-world applications, these models match job seekers with suitable employment opportunities based on resumes and job descriptions provided by employers. When handling new job descriptions, the process mirrors training by combining job descriptions and required skills into a single text field. The BERT model tokenizes the text and generates embeddings that capture the meaning of the text. These embeddings are then passed through the trained LSTM, MLP, and SVM models to predict the appropriate job title.

This chapter presents the development and testing of an enhanced job matching model using machine learning algorithms and BERT embeddings. It outlines the process of data acquisition and preparation, focusing on a comprehensive set of job descriptions, skills, and roles. Feature engineering plays a crucial role in enabling BERT to extract deep contextual embeddings of the input text, which are then used to train machine learning models such as LSTM networks, MLP, and SVM for performance optimization. The models are evaluated based on accuracy, providing insights into their predictive capabilities. The models' ability to predict job titles from new descriptions demonstrates their potential to improve recruitment processes, with the findings paving the way for future advancements in job matching using enhanced NLP models.

4. Results and Discussion

This chapter analyses the use of BERT embeddings with machine learning models for task matching, focusing on their accuracy and performance. It compares the strengths and weaknesses of the BERT classifier, LSTM, SVM, and MLP models, providing insights into their effectiveness in job title prediction and recruitment.

4.1 Result of BERT with Machine Learning Algorithms

This section examines the performance of ML algorithms augmented with BERT for matching, in terms of accuracy and context. As it is illustrated in Figure 3 that provides result of comparison of job matching models.

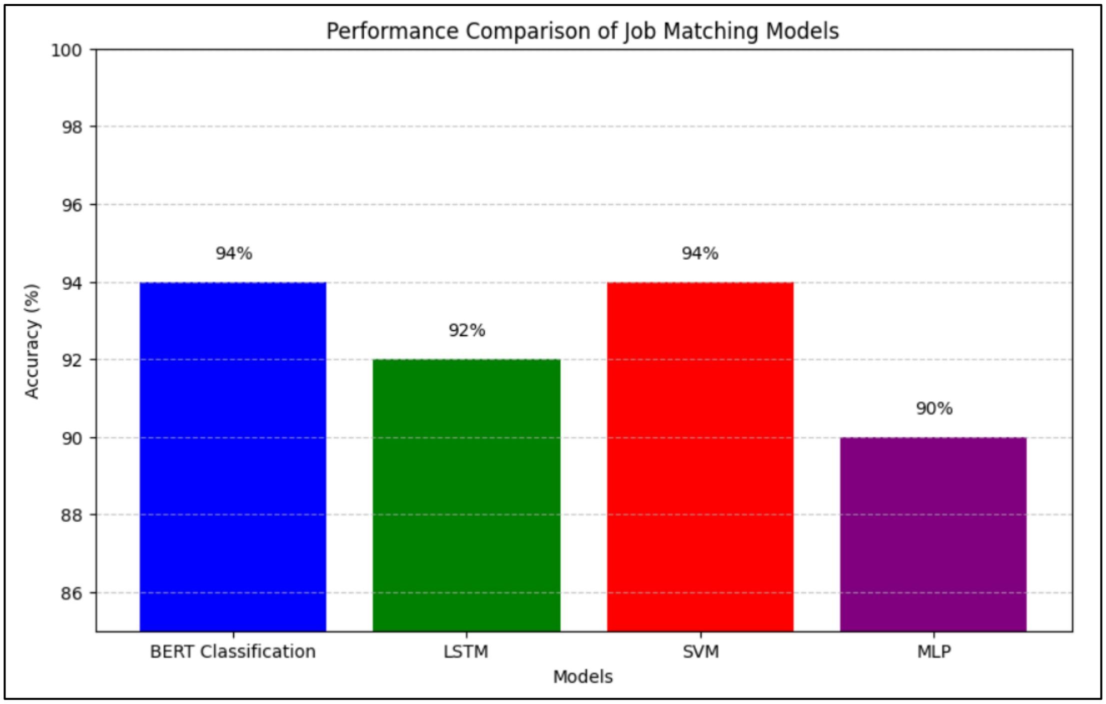

Figure 3: Performance Result.

The outcome included BERT Classification and Job Matching algorithms.

1. The BERT classification model achieved 94% accuracy, showcasing its ability to effectively identify semantic and contextual relationships in job descriptions, making it a highly reliable tool for task matching.

2. The LSTM model achieved 92% accuracy, demonstrating its effectiveness in handling sequential data and capturing long-term dependencies in job descriptions. While not as accurate as BERT, LSTM excels in interpreting trends and contexts over time.

3. The SVM model also achieved 94% accuracy, highlighting its ability to handle high-dimensional data and perform well in text categorization. When combined with BERT embeddings, SVM leverages rich contextual information, enhancing its performance in job matching tasks.

4. The MLP model achieved 90% accuracy, which is lower than the other models but still demonstrates its potential. MLPs are capable of recognizing complex patterns, though further adjustments are needed to enhance their performance and match the effectiveness of BERT and SVM.

4.2 Sample Prediction and analysis of job titles

This section presents a simulation of task predictions using machine learning models (LSTM, MLP, SVM) based on task descriptions. It evaluates the models' ability to classify job roles accurately, offering insights into their effectiveness in recruitment processes. The discrepancies between predicted and actual job titles highlight areas for model improvement and emphasize the challenges in capturing job activity nuances through machine learning. This analysis illustrates whether job titles match job descriptions effectively. In the Table 3, It has been shown that our job description allows us to assign job names based on the accuracy of the algorithm.

Table 3: Sample Prediction of Job Titles.

No.	Job Description	LSTM Predicted Job Title	MLP Predicted Job Title	SVM Predicted Job Title
1.	Backend Development (Python, Django, SQL, RESTful APIs)	UX/UI Designer	Back-End Developer	Web Developer
2.	Frontend Development (JavaScript, React, HTML, CSS)	UX/UI Designer	Mechanical Engineer	Web Developer
3.	Data Analysis (Python, R, SQL, Statistics)	UX/UI Designer	Mechanical Engineer	Project Coordinator

In the first case, the MLP model made the most accurate prediction, correctly identifying the job title as "Back-End Developer." The LSTM model, however, assigned the wrong title of "UX/UI Designer," indicating its difficulty in capturing the specific context of backend development. The SVM model predicted the title as "Web Developer," which, while somewhat related, was not entirely accurate. This suggests that while the SVM model can identify related areas, it may require more detailed information for precise job title prediction.

In the second case, the SVM model achieved the highest accuracy, correctly predicting the job title as "Web Developer." The LSTM model, however, predicted "UX/UI Designer," which is more related to front-end development but not entirely accurate, indicating a limited understanding of front-end and back-end competencies. The MLP model incorrectly predicted "Mechanical Engineer," showing a significant discrepancy and suggesting the need for more contextual data to improve accuracy in front-end tasks. Overall, the SVM model proved to be more efficient in recognizing front-end development roles.

In the third case, none of the models performed well in predicting the role based on the job description. The LSTM model predicted "UX/UI Designer," which was incorrect for data analysis, showing its limited ability to distinguish between technical functions. The MLP model misclassified the job title as "Mechanical Engineer," highlighting a significant mismatch in data analysis skills. The SVM model predicted "Project Coordinator," which, while somewhat relevant, was not entirely accurate, indicating the need for fine-tuning to better capture the contextual information of data analysis roles.

In summary, the LSTM model excelled at classifying "UX/UI Designer" but highlighted issues with ambiguity in job matching. The MLP model performed well for backend development but struggled with new roles, requiring more contextual data. The SVM model showed overall satisfactory results, successfully predicting frontend development tasks and demonstrating a strong awareness of current job trends in relevant fields.

4.3 Discussion

The integration of BERT embeddings significantly enhances the task matching algorithm's performance. Both BERT Classification and SVM models achieved an impressive 94% accuracy, demonstrating their ability to understand and interpret input data effectively. The LSTM model performed well with 92% accuracy, excelling in capturing sequential dependencies in job descriptions, while the MLP model showed promise with 90% accuracy, though it requires further optimization.

BERT Classification is particularly strong in semantic analysis, providing context and improving real-world job matching efficiency. LSTM excels at analyzing sequential data and long-term dependencies, though it struggles with broader context. SVM, when combined with BERT embeddings, is highly effective in text classification, ensuring accurate matching. MLP, while good at identifying complex patterns, needs further refinement to match the performance of BERT and SVM. Overall, BERT and SVM perform best, while MLP shows potential with more adjustments.

This study demonstrates the transformative potential of advanced NLP and ML techniques in recruitment, particularly by leveraging BERT embeddings for precise and context-aware job matching. By effectively capturing complex language structures, BERT and machine learning models, such as SVMs, reduce mismatches inherent in traditional keyword-based approaches, aligning job criteria with candidate qualifications. This enhanced accuracy streamlines hiring, saving time and resources for employers while improving job seeker satisfaction. Customizable and scalable, these models adapt to various industries, automate the matching process, and empower HR teams with data-driven insights, promoting fairness, inclusion, and efficiency in recruitment. The broader impact includes combating unemployment and fostering a more equitable workforce.

Despite promising results, this study has several limitations. The dataset used was specific and may not fully represent the broader labor market or other industries, leading to potential variability in results. Additionally, the complexity of advanced models like BERT and LSTM requires significant computational resources, limiting their accessibility for smaller projects. Lastly, while the models achieved high accuracy, further research is needed to address potential biases and ensure fairness, promoting equitable job matching and career opportunities. In summary, the combination of BERT embeddings and machine learning models significantly improved matching accuracy, with BERT and SVM achieving 94% accuracy and LSTM performing well due to its ability to process sequential data. While MLP demonstrated potential, it requires further refinement. These findings underscore the transformative impact of advanced NLP techniques in enhancing recruitment accuracy and efficiency, emphasizing the need for continued refinement of these models to optimize future job matching processes.

5. Conclusions and Future Directions

This chapter summarizes the main findings of our research to enhance the task matching process by incorporating BERT embeddings into other machine learning models examine the results of these findings, except the study limitations are acknowledged, and suggestions for further research are made. The project aimed to develop an advanced job matching algorithm using BERT embeddings and machine learning models. The findings demonstrated that integrating BERT with SVM, LSTM, and MLP models significantly enhanced the accuracy and efficiency of task matching. BERT embeddings improved contextual understanding, leading to more precise matches between job seekers and employers. The experimental results consistently surpassed traditional methods in task matching performance.

The key findings of the study include: the BERT and SVM models achieved an overall accuracy of 94%, demonstrating their effectiveness in analyzing the semantic networks within job descriptions and titles. The LSTM model reached 92% accuracy, highlighting its ability to handle sequential data and capture long-term dependencies crucial in job and candidate descriptions. Although the MLP model achieved an accuracy slightly above 89%, it shows potential for further refinement and improvement. Future research should focus on enhancing LSTM and MLP models by exploring advanced methods and training data to improve accuracy and performance. Additionally, investigating semi-supervised learning and more sophisticated deep learning architectures, such as attention mechanisms, hybrids, and ensemble methods, could capture complex data structures. Addressing bias detection and avoidance in job matching, to prevent biased hiring practices, is also crucial. Finally, industry-specific customization of models should be explored to ensure their feasibility and effectiveness in recruitment processes across various sectors.

The division of data into training, testing, and validation sets is crucial for hyperparameter tuning and model performance evaluation. Advanced deep learning techniques, such as hybrid sampling and clustering, can enhance model stability and uncover complex patterns in data. Additionally, addressing potential biases in job-matching models is essential to ensure fairness and inclusiveness, promoting equal employment opportunities across demographics. Tailoring models to specific industries and labor market characteristics can further optimize recruitment processes, improving both profitability and performance in the private sector.

References

Abdollahnejad, Elias, Marilynn Kalman, and Behrouz H Far. 2021. "A deep learning bert-based approach to person-job fit in talent recruitment." 2021 International Conference on Computational Science and Computational Intelligence (CSCI).

Ali, Irfan, Nimra Mughal, Zahid Hussain Khand, Javed Ahmed, and Ghulam Mujtaba. 2022. "Resume classification system using natural language processing and machine learning techniques." *Mehran University Research Journal of Engineering & Technology* 41 (1):65-79.

Alsaif, Suleiman Ali, Minyar Sassi Hidri, Hassan Ahmed Eleraky, Imen Ferjani, and Rimah Amami. 2022. "Learning-based matched representation system for job recommendation." *Computers* 11 (11):161.

Appadoo, Kevin, Muhammad Bilaal Soonnoo, and Zahra Mungloo-Dilmohamud. 2020. "Job recommendation system, machine learning, regression, classification, natural language processing." 2020 IEEE Asia-Pacific Conference on Computer Science and Data Engineering (CSDE).

Bhatia, Vedant, Prateek Rawat, Ajit Kumar, and Rajiv Ratn Shah. 2019. "End-to-end resume parsing and finding candidates for a job description using bert." *arXiv preprint arXiv:1910.03089*.

Bouhoun, Zakaria, Theo Guerrois, Xianli Li, Mouna Baker, Nassara Elhadji Ille Gado, Emir Roumili, Francesco Vitillo, Lies Benmiloud Bechet, and Robert Plana. 2023. "Information Retrieval Using Domain Adapted Language Models: Application to Resume Documents for HR Recruitment Assistance." International Conference on Computational Science and Its Applications.

Chala, Sisay, and Madjid Fathi. 2017. "Job seeker to vacancy matching using social network analysis." 2017 IEEE international conference on industrial technology (ICIT).

Chen, Xi. 2022. "Human Resource Matching Support System Based on Deep Learning." *Mathematical Problems in Engineering* 2022 (1):1558409.

Chou, Yi-Chi, and Han-Yen Yu. 2020. "Based on the application of AI technology in resume analysis and job recommendation." 2020 IEEE International Conference on Computational Electromagnetics (ICCEM).

Chowdhury, SM Shawal, Mrithika Chowdhury, and Arifa Sultana. 2022. "Matching Job Circular with Resume Using Different Natural Language Processing Based Algorithms." International Conference on Machine Intelligence and Emerging Technologies.

Delecraz, Sebastien, Loukman Eltarr, Martin Becuwe, Henri Bouxin, Nicolas Boutin, and Olivier Oullier. 2022a. "Making recruitment more inclusive: Unfairness monitoring with a job matching machine-learning algorithm." Proceedings of the 2nd International Workshop on Equitable Data and Technology.

Delecraz, Sebastien, Loukman Eltarr, Martin Becuwe, Henri Bouxin, Nicolas Boutin, and Olivier Oullier. 2022b. "Responsible Artificial Intelligence in Human Resources Technology: An innovative inclusive and fair by design matching algorithm for job recruitment purposes." *Journal of Responsible Technology* 11:100041.

Deng, Yu, Hang Lei, Xiaoyu Li, and Yiou Lin. 2018. "An improved deep neural network model for job matching." 2018 International Conference on Artificial Intelligence and Big Data (ICAIBD).

Deshmukh, Asmita, and Anjali Raut. 2024. "Applying BERT-Based NLP for Automated Resume Screening and Candidate Ranking." *Annals of Data Science*:1-13.

Devlin, Jacob, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. "Bert: Pre-training of deep bidirectional transformers for language understanding." *arXiv preprint arXiv:1810.04805*.

Faliagka, Evanthia, Kostas Ramantas, Athanasios Tsakalidis, and Giannis Tzimas. 2012. "Application of machine learning algorithms to an online recruitment system." Proc. International Conference on Internet and Web Applications and Services.

FraiJ, JihaD, and Várallyai László. 2021. "A literature review: artificial intelligence impact on the recruitment process." *International Journal of Engineering and Management Sciences* 6 (1):108-119.

Gadegaonkar, Sakshi, Darsh Lakhwani, Sahil Marwaha, and Abhijeet Salunke. 2023. "Job recommendation system using machine learning." 2023 Third International Conference on Artificial Intelligence and Smart Energy (ICAIS).

Green, Thomas Alexander Fleming. 2023. "Using NLP to resolve mismatches between jobseekers and positions in recruitment." University of Sheffield.

Guo, Shiqiang, Folami Alamudun, and Tracy Hammond. 2016. "RésuMatcher: A personalized résumé-job matching system." *Expert Systems with Applications* 60:169-182.

Hafizovic, Nedzad. 2019. Candidate-job recommendation system: Building a prototype of a machine learning–based recommendation system for an online recruitment company.

Homsapaya, Kanyanut, and Kridsada Budsara. 2020. "Machine learning for Older Jobseeker and Employment matching." 2020 17th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON).

Honorati, Maddalena, Céline Ferré, and Tomasz Gajderowicz. 2023. "Piloting a Machine Learning-Based Job-Matching Algorithm: Summary of Results from Pomerania."

Kaya, Mesut, and Toine Bogers. 2023. "An Exploration of Sentence-Pair Classification for Algorithmic Recruiting." Proceedings of the 17th ACM Conference on Recommender Systems.

Khan, Md Ashikur Rahman, Anjan Rhudra Paul, Fardowsi Rahman, Jony Akter, Zakia Sultana, and Masudur Rahman. 2023. "Appropriate Job Selection Using Machine Learning Techniques."

Kurek, Jarosław, Tomasz Latkowski, Michał Bukowski, Bartosz Świderski, Mateusz Łępicki, Grzegorz Baranik, Bogusz Nowak, Robert Zakowicz, and Łukasz Dobrakowski. 2024. "Zero-shot recommendation AI models for efficient job–candidate matching in recruitment process." *Applied Sciences* 14 (6):2601.

Liang, Fangning, and Xiangyu Wan. 2022. "Job Matching Analysis Based on Text Mining and Multicriteria Decision-Making." *Mathematical Problems in Engineering* 2022 (1):9245876.

Lin, Yiou, Hang Lei, Prince Clement Addo, and Xiaoyu Li. 2016. "Machine learned resume-job matching solution." *arXiv preprint arXiv:1607.07657*.

Lu, Yao, Sandy El Helou, and Denis Gillet. 2013. "A recommender system for job seeking and recruiting website." Proceedings of the 22nd International Conference on World Wide Web.

Martinez-Gil, Jorge, Bernhard Freudenthaler, and Thomas Natschläger. 2018. "Recommendation of job offers using random forests and support vector machines." Proceedings of the of the EDBT/ICDT joint conference.

Mostafa, Lamiaa, and Sara Beshir. 2021. "Job candidate rank approach using machine learning techniques." Advanced Machine Learning Technologies and Applications: Proceedings of AMLTA 2021.

Nigam, Amber, Aakash Roy, Hartaran Singh, and Harsimran Waila. 2019. "Job recommendation through progression of job selection." 2019 ieee 6th international conference on cloud computing and intelligence systems (ccis).

Osisanwo, FY, JET Akinsola, O Awodele, JO Hinmikaiye, O Olakanmi, and J Akinjobi. 2017. "Supervised machine learning algorithms: classification and comparison." *International Journal of Computer Trends and Technology (IJCTT)* 48 (3):128-138.

Panchasara, Sahil, Rajeev Kumar Gupta, and Ashutosh Sharma. 2023. "AI Based Job Recommedation System using BERT." 2023 7th International Conference On Computing, Communication, Control And Automation (ICCUBEA).

Pang, Ronghui. 2024. "Research on the Application of Word2Vec-Based Job-Person Matching Methods in Corporate Recruitment." Proceedings of the 2024 16th International Conference on Machine Learning and Computing.

Pendyala, Vishnu S, Nishtha Atrey, Tina Aggarwal, and Saumya Goyal. 2022. "Enhanced algorithmic job matching based on a comprehensive candidate profile using nlp and machine learning." 2022 IEEE Eighth International Conference on Big Data Computing Service and Applications (BigDataService).

Pombo, Luís Matos. 2019. "Landing on the right job: a machine learning approach to match candidates with jobs applying semantic embeddings." Universidade NOVA de Lisboa (Portugal).

Rahmani, Hamed, Wim Groot, and Amir Mohammad Rahmani. 2023. "A predictive analytics solution matching job seekers' talent and employers' demands based on machine learning."

Roy, Pradeep Kumar, Sarabjeet Singh Chowdhary, and Rocky Bhatia. 2020. "A Machine Learning approach for automation of Resume Recommendation system." *Procedia Computer Science* 167:2318-2327.

Sarker, Iqbal H. 2021. "Machine learning: Algorithms, real-world applications and research directions." *SN computer science* 2 (3):160.

Schwartz, Roy, Gabriel Stanovsky, Swabha Swayamdipta, Jesse Dodge, and Noah A Smith. 2020. "The right tool for the job: Matching model and instance complexities." *arXiv preprint arXiv:2004.07453*.

Singh, Amanpreet, Narina Thakur, and Aakanksha Sharma. 2016. "A review of supervised machine learning algorithms." 2016 3rd international conference on computing for sustainable global development (INDIACom).

Smelyakov, Kirill, Yuliia Hurova, and Serhii Osiievskyi. 2023. "Analysis of the effectiveness of using machine learning algorithms to make hiring decisions." *Proceedings http://ceur-ws. org ISSN* 1613:0073.

Sudha, G, KK Sasipriya, D Nivethitha, and S Saranya. 2021. "Personality prediction through CV analysis using machine learning algorithms for automated e-recruitment process." 2021 4th international conference on computing and communications technologies (ICCCT).

Tambe, Prasanna, Peter Cappelli, and Valery Yakubovich. 2019. "Artificial intelligence in human resources management: Challenges and a path forward." *California Management Review* 61 (4):15-42.

Tejaswini, K, V Umadevi, Shashank M Kadiwal, and Sanjay Revanna. 2022. "Design and development of machine learning based resume ranking system." *Global Transitions Proceedings* 3 (2):371-375.

Wang, Xiaowei, Zhenhong Jiang, and Lingxi Peng. 2021. "A Deep-Learning-Inspired Person-Job Matching Model Based on Sentence Vectors and Subject-Term Graphs." *Complexity* 2021 (1):6206288.

Yabin, Liu. 2022. "Research on human resource matching system based on collaborative filtering algorithm and trust propensity matching." 2022 IEEE 5th International Conference on Information Systems and Computer Aided Education (ICISCAE).

Zhou, Qing, Fenglu Liao, Chao Chen, and Liang Ge. 2019. "Job recommendation algorithm for graduates based on personalized preference." *CCF Transactions on Pervasive Computing and Interaction* 1:260-274.

Awlla, K.M., Veisi, H. and Abdullah, A.A., 2025. Sentiment analysis in low-resource contexts: BERT's impact on Central Kurdish. *Language Resources and Evaluation*, pp.1-31.

Abdullah, A.A., Abdulla, S.H., Toufiq, D.M., Maghdid, H.S., Rashid, T.A., Farho, P.F., Sabr, S.S., Taher, A.H., Hamad, D.S., Veisi, H. and Asaad, A.T., 2024. NER-RoBERTa: Fine-Tuning RoBERTa for Named Entity Recognition (NER) within low-resource languages. arXiv preprint arXiv:2412.15252.

Abdullah, A.A., Mohammed, N.S., Khanzadi, M., Asaad, S.M., Abdul, Z.K. and Maghdid, H.S., A Comprehensive Review of Machine Learning Approaches: Techniques, Applications, and Trends.