AI-Driven Professional Profile Categorization and Recommendation System

Marouane CHIHAB, Hicham BOUSSATTA, Mohamed CHINY, Nabil Mabrouk, Younes CHIHAB, Moulay Youssef HADI
Laboratory of Computer Sciences, Ibn Tofail University, Kenitra, Morocco

***Abstract*—The exponential growth of applications in digital and information system domains has made the identification of qualified candidates increasingly complex, resulting in longer and less efficient recruitment processes. Recruiters frequently deal with heterogeneous and unstructured résumés, which complicates skill assessment and increases the risk of mismatches between candidates and job requirements. To address these challenges, this research proposes an AI-based framework for the automatic classification and recommendation of professional profiles using natural language processing (NLP), text mining, and supervised machine learning techniques. The methodology includes the comparative evaluation of several classification algorithms—Logistic Regression, Random Forests, Support Vector Machines (SVM), k-Nearest Neighbors (KNN), Gradient Boosting (GB), and Naïve Bayes—to identify the most accurate and robust model. The framework also incorporates a similarity-based matching mechanism to align candidate profiles with job postings. Experimental results show a classification accuracy of 96.38%, demonstrating the model's effectiveness in enabling faster, more reliable, and objective recruitment decisions while providing candidates with insights into their compatibility with labor market expectations.**

Keywords*—Professional profile classification; profile recommendation; natural language processing (NLP); supervised learning; Logistic Regression; Random Forest; Support Vector Machine (SVM); k-Nearest Neighbors (KNN); Gradient Boosting; Naïve Bayes; AI-based recruitment

I. Introduction

Recruitment is currently undergoing a major transformation as a result of digitalization and the growing adoption of artificial intelligence. In a context marked by an exponential volume of applications, particularly in the technology sectors, companies are having to manage an increasing amount of unstructured textual data, mainly curriculum vitae (CV). These documents, although rich in information about career paths and professional skills, are problematically heterogeneous in terms of format, structure, and content, which makes systematic analysis difficult.

Traditionally, recruiters evaluate CVs manually, a method that has proved to be particularly limited: not only is this process time-consuming and costly in terms of resources, but it is also subject to human bias and ill-suited to processing large volumes of applications. Furthermore, the variability of the presentations, wording, and experience described makes it difficult to attempt to standardize and compare profiles objectively. On the candidate side, the lack of tools to understand the automated selection mechanisms and assess their suitability for the market is a further obstacle.

This research presents a comprehensive methodology for classifying professional profiles and assessing their match with job vacancies based on automated CV analysis. The approach is not limited to a technical solution, but establishes a reproducible, scalable, and interpretable methodological framework, combining text analysis, machine learning, and semantic similarity to provide results applicable to both recruiters and candidates.

The method consists of three main components: Firstly, the processing and conversion of textual data involves extracting the content of CVs (in PDF, Word, or text formats), purifying it by removing superfluous elements, normalizing it by reducing words to their root forms, and transforming it into numerical representations using techniques such as TF-IDF or semantic embeddings.

Secondly, the supervised learning classification phase is based on the training and comparative evaluation of several models (Logistic Regression, Random Forests, SVM, KNN, Gradient Boosting, and Naive Bayes) using cross-validation and rigorous metrics (precision, recall, F1-score). Thirdly, CV-offers are matched by calculating similarity (cosine or Euclidean distance), enabling them to be ranked by relevance and personalized recommendations to be generated.

There are many innovations in this work. On the one hand, it offers complete integration of the CV processing process, from import to final recommendation. Secondly, it is distinguished by a systematic comparison of traditional algorithms applied to the classification of CVs by profession - a problem that has been little explored to date. Finally, the emphasis is on producing actionable information: identifying key skills, relevant technologies, and gaps between candidate profiles and job requirements.

In concrete terms, the proposal significantly improves the efficiency of the selection process while reducing biases inherent in human assessment. For candidates, it provides a valuable tool to position themselves on the job market, enabling strategic adjustments to CVs and career paths.

From a research perspective, this study contributes to the state-of-the-art by proposing a reproducible methodology for the automated analysis and prediction of career paths based on CVs. This study addresses the following research question: How can an AI-based framework combining NLP and supervised machine learning, effectively classify professional

profiles and generate reliable job recommendations to improve recruitment efficiency?

The remainder of this study is structured as follows: Section II reviews related work, Section III presents the research methodology, Section IV reports and analyzes the experimental results, Section V offers the discussion, and Section VI provides a conclusion along with perspectives for future work.

II. Related Work

The continuing growth in the number of job seekers is leading to a significant increase in the volume of applications received for each vacancy advertised. Many of these applications are relevant to the post on offer. This creates a major problem for recruiters, who have to carry out a rigorous pre-selection process to identify the most suitable profiles [1, 2, 3]. The process of matching a candidate's curriculum vitae to a job description can be compared to the operation of a referral system, in which an individual's profile is proposed for a given job. The notion of a recommendation system was introduced by Resnick and Varian [4], and has since been widely democratized in many fields of application. These systems are now ubiquitous, particularly in the context of product recommendation on e-commerce platforms [5, 6], but also in services for suggesting books [7], press articles [8], films [9], music content [10], and in many other contexts [11, 12, 13, 14, 15].

Lu et al [12] proposed an in-depth review of the protocols adopted over the years in the field of recommender systems. Their study highlights the evolution of these systems and their increasing integration into real-time applications. In addition, Wei et al [16] carried out a detailed analysis of the various recommendation techniques, outlining the fundamental principles underlying their operation. For their part, Al-Otaibi et al [17] were specifically interested in recommender systems applied to the employment domain. Their work presents a complete overview of the recruitment process, from the identification of needs to the final selection of candidates. In particular, they explain how online recruitment portals make it easier to match vacancies with profiles, and identify the key factors that can influence a candidate's selection decision, while detailing the different stages of the HR process implemented by organizations.

Paparrizos et al [13] have proposed an innovative hybrid classifier-based model for the job recommendation system. Their approach combines several components, including information retrieval techniques, manually defined attributes, and other relevant indicators, in order to improve the relevance of recommendations and optimize the match between candidate profiles and job offers.

The application process is often a demanding and tedious stage for job seekers. However, CV optimization is an essential part of this process, as it enables candidates to effectively showcase their skills and experience to potential recruiters [18]. In this context, machine learning algorithms are playing an increasingly important role. These increasingly sophisticated algorithms are capable of constantly learning from new data, user feedback and changes in the job market. Their adaptability allows them to analyze CV performance over time and iteratively adjust their recommendation and optimization mechanisms in response to feedback [19].

In artificial intelligence algorithms applied to recruitment, CVs and job descriptions are processed by machine learning models to analyze and extract relevant information. These models enable automated matching between candidate profiles and job offers, based on various criteria such as keywords, semantic analysis, and other contextual indicators [20].

In [21], the authors explored in depth the joint application of automatic natural language processing (NLP) techniques and machine learning algorithms in the context of CV analysis and optimization. Their approach aims to automatically adapt CVs to the specific requirements of various positions, taking into account the skills required and the criteria implicit in the job offers. This process leads to the generation of optimized CVs, promoting a better match between the candidate's profile and recruiters' expectations. As a result, this method significantly improves recall rates and the chances of selecting candidates.

As part of the development of predictive models, several widely used supervised learning algorithms—such as Logistic Regression, Support Vector Machines (SVM), Random Forests, Gradient Boosting, k-Nearest Neighbors (k-NN), and Naive Bayes classifiers—were compared in order to evaluate their respective performances. The primary objective is to develop a decision-support tool for recruiters, enabling them to efficiently preselect candidates whose profiles best match the specific requirements of open positions. This approach seeks to enhance the quality of recruitment by improving both hiring satisfaction and overall organizational productivity.

In conclusion, this study represents a significant contribution to existing research by introducing a method that is both rigorous and easily reproducible for the analysis, classification, and prediction of professional career paths based on CVs. Unlike previous approaches, which often rely on manual feature extraction, limited workflows, or models with lower accuracy, the proposed AI-based framework employs a different and fully integrated approach, combining text mining, multiple supervised learning algorithms, and a similarity-based recommendation mechanism. This method achieves a classification accuracy of 96.38%, demonstrating higher performance, greater automation, and better adaptability compared to existing models, thus providing a comprehensive solution tailored to the needs of the job market.

III. Research Methodology

In this work, the methodological approach is based on a rigorous automated processing pipeline for CVs, encompassing every stage from data acquisition to the generation of recommendations. The textual data underwent an in-depth preprocessing stage. The CVs, collected in various formats (PDF, DOCX, LinkedIn), were subjected to a series of transformations, including lowercasing, punctuation and stop-word removal, and advanced lemmatization. This crucial phase ensures standardization of the corpus and prepares the data for subsequent analysis.

The next phase concerns the vector representation of the texts. The TF-IDF method was primarily employed to transform the CVs into exploitable numerical data, while future enhancements using embedding techniques such as Word2Vec or BERT are considered. Particular attention was given to class balancing using the Random Over Sampler method, ensuring optimal conditions for the machine learning processes.

The core idea of the approach lies in the comparative implementation of several supervised classification algorithms. Six models were systematically evaluated: Logistic Regression, Support Vector Machines (SVM), Random Forest, Gradient Boosting, k-Nearest Neighbors (k-NN), and Naive Bayes. Each model is tested following rigorous protocols, including cross-validation and hyperparameter optimization. Their performance is assessed using standardized metrics (accuracy, F1-score, recall), enabling an objective selection of the most effective model.

Finally, the system computes cosine similarity scores between CVs and job descriptions, thereby establishing an automatic ranking of applications. This final phase incorporates weighting mechanisms to account for key skills and job-specific requirements. The entire pipeline has been architected to ensure reproducibility, scalability, and seamless integration with existing talent management systems.

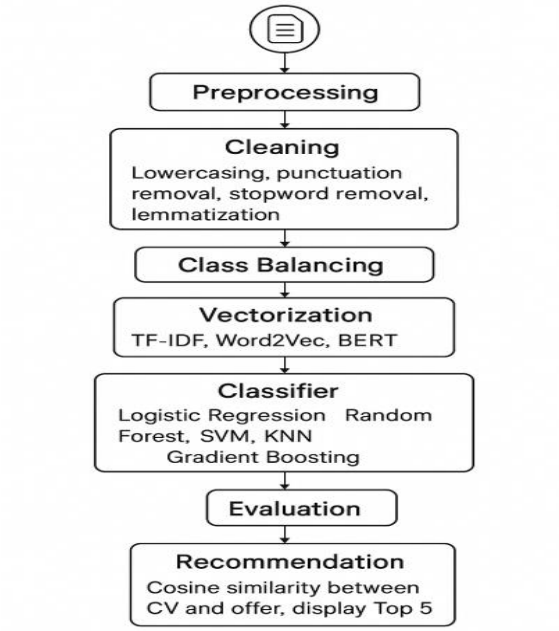

Fig. 1. Framework of the proposed model.

Fig. 1 illustrates the entirety of the approach, from the data collection phase to the generation of final predictions and recommendations.

1) Extracting CVs: Extraction is the first essential stage in the automated CV processing process. It involves collecting documents from a variety of sources, such as files in PDF or DOCX format or LinkedIn profiles, representing the wide range of formats used in modern job applications. In order to standardize this heterogeneous data, the documents are converted to plain text using specialized libraries such as pdf miner, docx2txt, or text extract, which are commonly used to extract textual content from unstructured files in a reliable and structured manner [22]. The aim of this transformation is to standardize the information, an essential prerequisite for any automatic processing by NLP algorithms. In particular, it makes it possible to recover the key elements present in a CV, such as professional experience, technical skills, diplomas obtained, as well as other metadata useful for semantic analysis and supervised classification. By guaranteeing accurate and complete extraction, this stage lays the foundations for a robust processing pipeline, capable of operating on a wide range of sources and formats while ensuring maximum fidelity to the initial information.

In this context, a rich and heterogeneous dataset of 1,428 CVs was constructed, collected from students with a wide range of profiles via the LinkedIn page. These documents, in multiple formats (PDF, Word, plain text), reflect significant structural and semantic diversity, representing a realistic dataset. This database has enabled us to explore the challenges of automatic CV processing in a context of large volumes and syntactic variability, offering an ideal terrain for the application of advanced natural language processing (NLP) techniques.

The figure below (see Fig. 2) illustrates the distribution of CVs according to their professional category, providing a valuable overview of the dominance of different technical profiles in the sample studied. It is an exploded pie chart, allowing not only the relative proportions to be visualized but also the dominant categories to be highlighted by a clear visual separation.

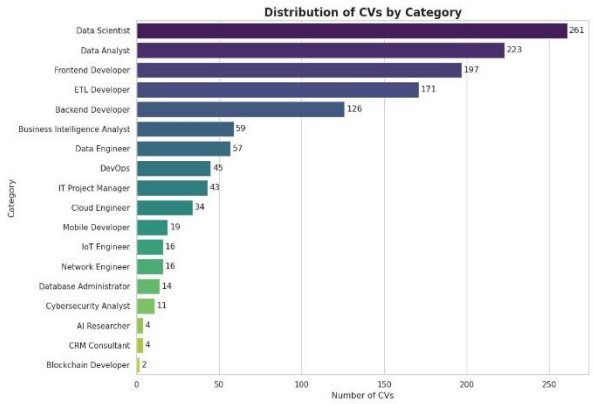

Fig. 2. Distribution of CVs by category.

2) Text pre-processing: Once the texts have been extracted, they undergo rigorous linguistic pre-processing to improve their quality and relevance for the subsequent phases of automatic analysis. This process involves a number of successive operations designed to normalize the textual data and reduce its noise. The first step is to convert all characters to lower case, so that words are treated uniformly regardless of their case. Next, punctuation, irrelevant numbers and special characters are removed to eliminate non-informative

elements. This type of text processing is fundamental to NLP, as it enables a coherent, relevant corpus to be built up that can be used by machine learning models. These operations, based on the recommendations of Bird et al [23], are now standard practice in the processing of complex textual data such as CVs.

3) Class balancing: In CV datasets, there is often a marked imbalance between the different occupational classes. This imbalance can lead to a bias in the machine learning models, which will tend to favor the majority classes to the detriment of the minority ones, thus compromising the quality of the predictions. To remedy this problem, a resampling strategy is implemented, in particular via the Random Over Sampler algorithm, which consists of randomly duplicating the samples of under-represented classes in order to balance the overall distribution of the data. This method, inspired by the SMOTE approach developed by Chawla et al [24], makes it possible to increase the representativeness of minority classes without introducing complex artificial information, and facilitates better generalization of the model, particularly for occupations rarely.

4) Vectorization of texts: Once the text has been cleaned, it is transformed into numerical vectors so that it can be used by machine learning algorithms. The main method used is TF-IDF (Term Frequency-Inverse Document Frequency), which assigns a weight to each word according to its frequency in a given document compared with its frequency in the corpus as a whole. This technique highlights the characteristic terms of a document while reducing the influence of over-frequent words, and is a benchmark approach to text vectorization [25,26,27,28,29,30]. However, this method remains purely statistical and does not incorporate semantic or contextual information. To enrich the representation of texts, experiments have therefore been carried out with Word2Vec, a lexical folding model developed by Mikolov et al [31], which captures the semantic relations between words by learning continuous representations in a vector space. With a view to improvement, the use of more advanced models such as BERT (Bidirectional Encoder Representations from Transformers), capable of producing rich and dynamic contextual embeddings, is envisaged to significantly increase the quality of text comprehension [32].

5) Training supervised models: Once the texts have been vectorized, they are used to train several supervised classification models with the aim of predicting the job category associated with a CV. Six algorithms were selected to cover a wide range of training strategies: Logistic Regression, the Support Vector Machine (SVM) recommended by Cortes & Vapnik [33] for its robustness in high-dimensional spaces, Random Forest, k-Nearest Neighbors (k-NN), Gradient Boosting, introduced by Friedman [34] for its effectiveness in complex tasks, and Naive Bayes, often used for its simplicity and speed of execution. Each model is trained by applying a cross-validation procedure, which makes it possible to adjust the hyperparameters, prevent overlearning and ensure good generalization capacity on new data. This algorithmic diversity makes it possible to compare different approaches, ranging from linear and interpretable models to more powerful sets of trees, as recommended in the good practice in machine learning documented by Pedregosa et al [35,36,37,38].

Six classification algorithms were used in this work, namely: Logistic Regression (LR), k-Nearest Neighbors (k-NN), Random Forest (RF), Support Vector Machine (SVM), Gradient Boosting, and Naive Bayes (NB). Table I summarizes the chosen hyper-parameters, which were then applied to the classification models. These hyper-parameters were selected experimentally, taking those that provided the best possible evaluations for the dataset.

TABLE I HYPER-PARAMETERS APPLIED TO ALGORITHMS IMPLEMENTED IN CLASSIFICATION MODELS

Algorithm	***Parameters***
Logistic Regression	max_iter=2000, class_weight='balanced'
Random Forest	n_estimators=100, max_depth=10
SVM	kernel='linear', class_weight='balanced'
KNN	n_neighbors=3
Gradient Boosting	n_estimators=100, learning_rate=0.1
Naive Bayes	alpha=1.0

6) Performance assessment: Model performance is rigorously evaluated using an independent test dataset, based on four fundamental machine learning metrics: accuracy (overall rate of correct predictions), precision (proportion of true positives among positive predictions), recall (ability to correctly identify all relevant cases), and the F1-score, which represents the harmonic mean between precision and recall. These indicators allow us to measure not only the overall effectiveness of the model, but also its ability to treat all classes fairly, in particular the minority occupational categories that are often under-represented. This multi-criteria evaluation approach, recommended in the systematic analysis by Sokolova & Lapalme [39], guarantees a complete and balanced assessment of the models, taking into account the different aspects of performance depending on the context of application.

7) Recommending candidates: Finally, the best performing model is integrated into an automated recommendation system, designed to assess the relevance of applications in relation to a specific job offer. To do this, the text of the vacancy and that of each CV are vectorized using the same text representation method as that used during training (TF-IDF in particular). A cosine similarity is then calculated between the vectors of the offer and those of the CVs, making it possible to measure their degree of semantic correspondence [40]. This technique, well documented in the reference work by Manning et al [41], produces a similarity score that reflects the lexical and contextual proximity

between two documents. On the basis of these scores, the system automatically ranks the CVs according to their suitability for the target job, and displays the five most relevant profiles, thus facilitating the work of recruiters by offering them an objective, rapid and justified pre-selection.

IV. Results

The performance of the trained models was measured using four main metrics: accuracy, precision, recall and F1-score. The data was divided into training (80%) and test (20%) sets, allowing a realistic assessment of performance.

The six models tested were: Logistic Regression, Random Forest, SVM, k-Nearest Neighbors, Gradient Boosting, and Naive Bayes. Each model was trained with specific hyperparameters, which were manually adjusted to guarantee the stability of the results.

TABLE II The Performance Measures of our Module Obtained by the Implementation of the Six Classification Algorithms

Model	*Accuracy*	*Precision*	*Recall*	*F1-Score*
Logistic Regression	94.68 %	94.63 %	94.77 %	94.60 %
Random Forest	89.57 %	89.37 %	89.83 %	88.95 %
SVM (linéaire)	96.38 %	96.42 %	96.53 %	96.40 %
k-NN	73.51 %	89.83 %	74.64 %	73.23 %
Gradient Boosting	94.79 %	94.73 %	95.02 %	94.79 %
Naive Bayes	80.96 %	82.76 %	81.98 %	80.20 %

As shown in Table II, the comparative evaluation of the six supervised classification models on a corpus of vectorized CVs using the TF-IDF method highlights clear trends in their respective effectiveness. The entire dataset, previously balanced using oversampling techniques (Random Over Sampler), was divided into 80% for training and 20% for testing, thus guaranteeing a robust estimate of performance on unseen data. Of the models tested, the Support Vector Machine (SVM) clearly emerged as the best performer, achieving an accuracy of 96.38%, a precision of 96.42%, a recall of 96.53% and an F1-score of 96.40%. These results confirm the ability of the SVM, recommended by Cortes & Vapnik [42], to manage high-dimensional text data, maintaining an excellent balance between class detection (recall) and prediction accuracy (precision). Right behind them, the Gradient Boosting (94.79% accuracy) and Logistic Regression (94.68%) models also offer excellent compromises, thanks in particular to their stability and low sensitivity to noisy variables. On the other hand, the performance of the k-Nearest Neighbors (k-NN) is more mixed: although it achieves high accuracy (89.83%), its results on recall (74.64%) and F1-score (73.23%) reflect a difficulty in generalizing effectively, probably due to its dependence on distances in high-dimensional vector spaces, which makes this model less suitable in the NLP context. As for Naive Bayes Multinomial, its overall performance was modest (80.96% accuracy), confirming that its assumptions of strong independence between variables can be limiting when it comes to capturing semantic dependencies in text. The Random Forest, although usually robust, lags slightly behind here with an accuracy of 89.57%, probably penalized by the complexity of the lexical structures. These results underline the importance of choosing models adapted to the specific nature of the textual data, and confirm that methods such as SVM and Gradient Boosting, when properly calibrated, can significantly outperform more traditional approaches in fine-grained, contextual classification tasks such as CVs.

The figure below (see Fig. 3) presents a comparative analysis of the performance of six classification algorithms (Logistic Regression, Random Forest, Linear SVM, k-NN, Gradient Boosting, and Naive Bayes) using four standard metrics: accuracy, precision, recall, and F1 score. Each sub-graph highlights the variations in performance according to the algorithm for a given metric, providing a clear and structured view of the strengths and weaknesses of each model. This visualization is intended to guide the choice of the most suitable model according to the priority criterion of the analysis.

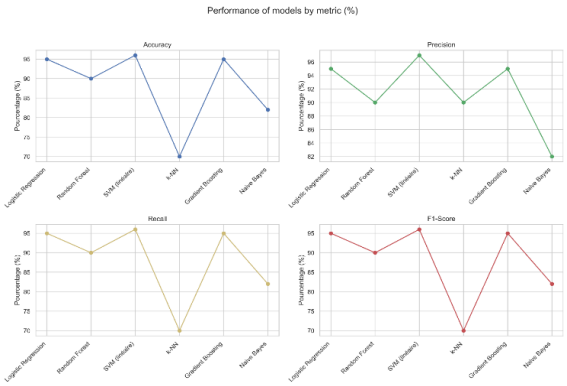

Fig. 3. Detailed comparison of model performance by metric.

The figure below (see Fig. 4) illustrates the comparative performance of the six classification algorithms studied, namely Logistic Regression, Random Forest, Linear SVM, k-Nearest Neighbors, Gradient Boosting, and Naive Bayes Multinomial. Four key metrics are represented: accuracy, precision, recall, and F1-score. These indicators are used to assess the overall quality of the models, taking into account both their ability to correctly predict positive and negative classes. This visualization highlights the differences in performance between the models and makes it easier to understand their respective strengths and limitations in the context of the classification task being analyzed.

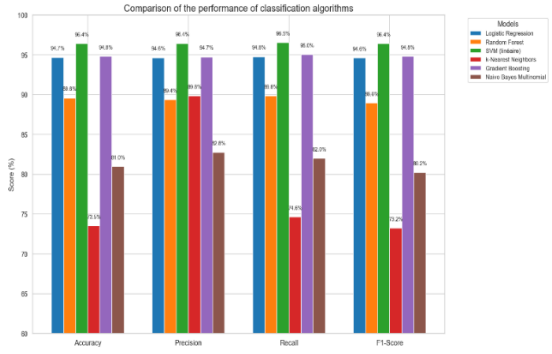

Fig. 4. Comparison of the performance of six classification algorithms according to four key metrics.

V. Discussion

The comparative evaluation of model performances enabled us to draw key and decisive conclusions for the model selection process and the definition of future optimization directions. The experimental results reveal significant differences among the tested models, both in terms of overall accuracy and their ability to balance recall and precision.

The linear SVM model clearly stands out as the best performer, achieving an accuracy of 96.38%, a precision of 96.42%, a recall of 96.53%, and an F1-score of 96.40%. These high values demonstrate that the SVM not only excels at correctly identifying positive instances but also effectively limits errors, indicating excellent generalization on the dataset. These results corroborate prior studies [42, 43, 44], which highlight the effectiveness of SVMs for high-dimensional classification problems, particularly when classes are linearly or nearly linearly separable.

The Gradient Boosting and Logistic Regression models achieve comparable performances, although slightly lower than those of the linear SVM. Gradient Boosting attains an accuracy of 94.79%, a precision of 94.73%, a recall of 95.02%, and an F1-score of 94.79%, compared to an accuracy of 94.68%, a precision of 94.63%, a recall of 94.77%, and an F1-score of 94.60% for

Logistic Regression. These results confirm their robustness:

- Logistic Regression remains relevant in binary classification problems, notably due to its interpretability and direct modeling of probabilities (Hosmer et al., 2013).
- Gradient Boosting, by sequentially optimizing residual errors [45, 46], combines flexibility and bias reduction, which explains its competitive performance.

Lagging behind the previous models, the Random Forest achieves an accuracy of 89.57%, a precision of 89.37%, a recall of 89.83%, but a lower F1-score of 88.95%. This slight discrepancy among the metrics indicates that the model effectively detects true positives while struggling to maintain an optimal balance between false positives and false negatives, which can impact its overall effectiveness. Although robust and widely used, Random Forest can sometimes perform less well on certain specific datasets due to its decision mechanism based on majority voting among trees, which may lead to dilution of optimal decisions (Breiman, 2001).

The k-Nearest Neighbors (k-NN) model exhibits mixed performance, with a modest accuracy of 73.51% and a low F1-score of 73.23%, despite a high precision of 89.83%. This imbalance between precision and recall (74.64%) reveals a tendency to under-detect true positives, due to:

- KNN's sensitivity to noise: The nearest neighbors may include outliers, which can distort the prediction.
- The curse of dimensionality effect: In high-dimensional spaces, the Euclidean distance (Cover & Hart, 1967) loses its relevance, diluting the notion of "closeness".

These limitations, consistent with the literature, suggest that k-NN is poorly suited for this dataset.

The Multinomial Naive Bayes model, with an accuracy of 80.96%, a precision of 82.76%, a recall of 81.98%, and an F1-score of 80.20%, achieves acceptable but clearly lower results compared to the top-performing models. This more modest performance is consistent with the fundamental assumption of conditional naïve independence among features, which is often violated in real-world data [47, 48].

In summary, the best-performing models achieve a close balance between precision and recall, as reflected by F1 scores near both metrics, while less effective models show greater imbalance, limiting reliable classification. These results highlight that linear SVM, Gradient Boosting, and Logistic Regression are the most effective choices, balancing performance, robustness, and interpretability. Moreover, combining these models with the text-mining preprocessing and the similarity-based recommendation mechanism enhances overall system effectiveness, providing actionable insights into candidate-job alignment. Building on these findings, future work can explore additional model architectures, deeper feature extraction, and dynamic recommendation strategies to further improve classification accuracy and applicability across diverse datasets.

VI. Conclusion

In this work, a comprehensive system for CV analysis and recommendation was presented, combining advanced NLP techniques and machine learning to address the challenges of modern recruitment. Experiments demonstrated the effectiveness of a hybrid approach, integrating both a supervised classification phase and a recommendation step based on semantic similarity. The results obtained, with an accuracy reaching 96.38% for the linear SVM, confirm the relevance of this approach and align with previous studies in the field.

The proposal also enabled the design, development, and evaluation of a comprehensive system for CV analysis, classification, and recommendation, based on advanced techniques in natural language processing (NLP), supervised machine learning, and interactive visualization through Streamlit.

To address the problem of manual CV sorting, an automated model based on machine learning was proposed, capable of recommending the most relevant applications to recruiters from a job description. The system operates in two phases: first, CVs are classified into different categories using supervised classifiers; then, profiles are recommended based on a semantic similarity index between the content of the CV and that of the job offer. This approach effectively captures the textual and semantic information present in the CVs.

The architecture of the proposed system is modular and scalable, structured around a robust pipeline. It integrates all

essential steps: text extraction, linguistic cleaning, TF-IDF vectorization, class balancing, classification (SVM, Random Forest, Gradient Boosting), and similarity computation for recommendation. This processing chain has been implemented to ensure system reproducibility and to provide a flexible framework for future developments.

The experimental results obtained confirm the validity of the proposed system. The SVM model, combined with TF-IDF vectorization, demonstrated the best performance with an accuracy of 96.38%. Other models such as Gradient Boosting and Logistic Regression also yielded satisfactory results, while the performances of Naive Bayes and k-NN remained limited in a complex semantic context.

One of the major contributions of this research work also lies in the integration of an interactive interface via Streamlit. This interface enables HR professionals, even without technical expertise, to upload CVs, visualize classification results, examine compatibility scores, and interact with various stages of the process.

In conclusion, this work has demonstrated the feasibility of building an automated, intelligent, and efficient solution to assist the recruitment process. The proposal constitutes a scientifically validated proof of concept, providing a solid foundation for the future industrialization of this type of approach. By combining methodological rigor, technical performance, and user accessibility, the approach contributes to advancing the use of artificial intelligence in modern talent management.

REFERENCES

[1] Breaugh, J. A. "The use of biodata for employee selection: Past research and future directions". Human Resource Management Review, 19, 219–231, (2009).

[2] Zhang, L., Fei, W., & Wang, L. "PJ Matching Model of Knowledge Workers". Procedia Computer Science, 60, 1128–1137, (2015).

[3] Roy, P. K., Singh, J. P., Baabdullah, A. M., Kizgin, H., & Rana, N. P. "Identifying reputation collectors in community question answering (CQA) sites: Exploring the dark side of social media". International Journal of Information Management, 42, 25–35, (2018).

[4] Resnick, P., & Varian, H. R. "Recommender systems". Communications of the ACM, 40(3), 56–59, (1997).

[5] Schafer, J. B., Konstan, J., & Riedl, J. "Recommender systems in e-commerce". Proceedings of the 1st ACM Conference on Electronic Commerce, 158–166, (1999).

[6] Singh, J. P., Irani, S., Rana, N. P., Dwivedi, Y. K., Saumya, S., & Roy, P. K. "Predicting the helpfulness of online consumer reviews". Journal of Business Research, 70, 346–355, (2017).

[7] Mooney, R. J., & Roy, L. "Content-based book recommending using learning for text categorization". Proceedings of the Fifth ACM Conference on Digital Libraries, 195–204, (2000).

[8] Das, A. S., Datar, M., Garg, A., & Rajaram, S. "Google news personalization: scalable online collaborative filtering". Proceedings of the 16th International Conference on World Wide Web, 271–280, (2007).

[9] Diao, Q., Qiu, M., Wu, C. Y., Smola, A. J., Jiang, J., & Wang, C. "Jointly modeling aspects, ratings and sentiments for movie recommendation (JMARS)". Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 193–202, (2014).

[10] Celma, O. "Music recommendation". In Music Recommendation and Discovery, Springer, 43–85, (2010).

[11] Carrer-Neto, W., Hernández-Alcaraz, M. L., Valencia-García, R., & García-Sánchez, F. "Social knowledge-based recommender system". (2012).

[12] Lu, J., Wu, D., Mao, M., Wang, W., & Zhang, G. "Recommender system application developments: A survey". Decision Support Systems, 74, 12–32, (2015).

[13] Paparrizos, I., Cambazoglu, B. B., & Gionis, A. "Machine-learned job recommendation". Proceedings of the Fifth ACM Conference on Recommender Systems, 325–328, (2011).

[14] Yi, X., Allan, J., & Croft, W. B. "Matching resumes and jobs based on relevance models". Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, 809–810, (2007).

[15] Roy, P. K., & Singh, J. P. "A Tag2Vec Approach for Questions Tags Suggestion on Community Question Answering Sites". International Conference on Machine Learning and Data Mining in Pattern Recognition, 168–182, (2018).

[16] Wei, K., Huang, J., & Fu, S. "A survey of e-commerce recommender systems". Proceedings of the 2007 International Conference on Service Systems and Service Management, 1–5, (2007).

[17] Al-Otaibi, S. T., & Ykhlef, M. "A survey of job recommender systems". International Journal of Physical Sciences, 7, 5127–5142, (2012).

[18] Kulkarni, A., Shankarwar, T., & Thorat, S. "Personality Prediction Via CV Analysis using Machine Learning". International Journal of Engineering Research & Technology, 10(9), (2021). Available from: https://www.ijert.org/research/personality-prediction-via-cv-analysis-using-machine-learning-IJERTV10IS090197.pdf.

[19] Kumar, A., et al. "Résumé Ranking and Selection using Machine Learning Algorithms". Expert Systems with Applications, 95, 283–298, (2018).

[20] Kaur, G., & Maheshwari, S. "Personality Prediction through Curriculum Vitae Analysis involving Password Encryption and Prediction Analysis". International Journal of Advanced Science and Technology, 28(16), 1–10, (2019).

[21] Liu, Y., Li, S., & Han, D. "Intelligent Resume Parsing Method Based on Deep Learning". Proceedings of the 2020 IEEE 2nd International Conference on Computer Science and Artificial Intelligence (CSAI), 628–632, (2020).

[22] Jain, A., & Sharma, S. "Text Extraction from Unstructured Resume Data for Candidate Selection Using Python Libraries". Procedia Computer Science, 167, 1741–1750, (2020).

[23] Bird, S., Klein, E., & Loper, E. "Natural Language Processing with Python". O'Reilly Media, (2009).

[24] Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. "SMOTE: Synthetic Minority Over-sampling Technique". Journal of Artificial Intelligence Research, 16, 321–357, (2002).

[25] Ramos, J. "Using TF-IDF to Determine Word Relevance in Document Queries". Proceedings of the First Instructional Conference on Machine Learning, (2003).

[26] Kang, G., Tang, M., Liu, J., Liu, X., & Cao, B. "Diversifying web service recommendation results via exploring service usage history". IEEE Transactions on Services Computing, 9, (2016).

[27] Guo, A., & Yang, T. "Research and improvement of feature words weight based on TF-IDF algorithm". Proceedings of the 2016 IEEE Information Technology, Networking, Electronic and Automation Control Conference, 415–419, (2016).

[28] Shengq, W., Huaizhen, K., Chao, L., Wanli, H., Lianyong, Q., & Hao, W. "Service Recommendation with High Accuracy and Diversity". Wireless Communications and Mobile Computing, (2020).

[29] Chiny, M., Chihab, M., Bencharef, O., & Chihab, Y. "LSTM, VADER and TF-IDF based Hybrid Sentiment Analysis Model". International Journal of Advanced Computer Science and Applications (IJACSA), 12(7), (2021). http://dx.doi.org/10.14569/IJACSA.2021.0120730.

[30] Chiny, M., Chihab, M., Bencharef, O., & Chihab, Y. "Netflix Recommendation System based on TF-IDF and Cosine Similarity Algorithms". Conference Paper, (2022). DOI: 10.5220/0010727500003101.

[31] Mikolov, T., Chen, K., Corrado, G., & Dean, J. "Efficient Estimation of Word Representations in Vector Space". arXiv:1301.3781, (2013).

[32] Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding". NAACL-HLT, (2019).

[33] Cortes, C., & Vapnik, V. "Support-vector networks". Machine Learning, 20(3), 273–297, (1995).

[34] Friedman, J. H. "Greedy Function Approximation: A Gradient Boosting Machine". Annals of Statistics, 29(5), 1189–1232, (2001).

[35] Pedregosa, F., et al. "Scikit-learn: Machine Learning in Python". Journal of Machine Learning Research, 12, 2825–2830, (2011).

[36] Chihab, M., et al. "BiLSTM and Multiple Linear Regression based Sentiment Analysis Model using Polarity and Subjectivity of a Text". International Journal of Advanced Computer Science and Applications, 13(10), (2022). DOI:10.14569/IJACSA.2022.0131052.

[37] Chihab, Y., Bousbaa, Z., Chihab, M., Bencharef, O., & Ziti, S. "Algo-Trading Strategy for Intraweek Foreign Exchange Speculation Based on Random Forest and Probit Regression". Applied Computational Intelligence and Soft Computing, (2019). https://doi.org/10.1155/2019/8342461.

[38] Boussatta, H., Chihab, M., Chiny, M., & Chihab, Y. "Predicting Oil Price Trends During Conflict With Hybrid Machine Learning Techniques". Applied Computational Intelligence and Soft Computing, (2020). https://doi.org/10.1155/acis/8867520.

[39] Yunxiang, L., Qi, X., & Zhang, T. "Research on Text Classification Method based on TF-IDF and Cosine Similarity". Journal of Information and Communication Engineering, 6(1), 335–338, (2020).

[40] Sokolova, M., & Lapalme, G. "A Systematic Analysis of Performance Measures for Classification Tasks". Information Processing & Management, 45(4), 427–437, (2009).

[41] Manning, C. D., Raghavan, P., & Schütze, H. "Introduction to Information Retrieval". Cambridge University Press, (2008).

[42] Gomez, J., Alfaro, C., Ortega, F., et al. "Adapting Support Vector Optimisation Algorithms to Textual Gender Classification." TOP, vol. 32, pp. 463–488, 2024.

[43] Rath, S. K., Sahu, M., Das, S. P., Bisoy, S. K., & Sain, M. "A Comparative Analysis of SVM and ELM Classification on Software Reliability Prediction Model." Electronics, vol. 11, no. 17, 2707, 2022.

[44] Kharoubi, R., Mkhadri, A., & Oualkacha, K. "High-Dimensional Penalized Bernstein Support Vector Machines." arXiv preprint, arXiv:2303.09066, 2023.

[45] Chen, T., & Guestrin, C. "XGBoost: A Scalable Tree Boosting System." Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD), pp. 785–794, 2016.

[46] Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., & Liu, T.-Y. "LightGBM: A Highly Efficient Gradient Boosting Decision Tree." Advances in Neural Information Processing Systems (NeurIPS), vol. 30, 2017.

[47] Rish, I. "An Empirical Study of the Naive Bayes Classifier." IJCAI 2001 Workshop on Empirical Methods in AI, 2001.

[48] Zhang, H. "The Optimality of Naive Bayes." Proceedings of the 17th International Conference on Knowledge Discovery and Data Mining, 2020.