

# Improved Candidate-Career Matching Using Comparative Semantic Resume Analysis

Asrar Hussain Alderham<sup>\*</sup>, Emad Sami Jaha

Faculty of Computer and Information Technology, Department of Computer Science, King Abdulaziz University, Jeddah, 22233, Saudi Arabia

## ARTICLE INFO

### Article history:

Received: 12 November, 2023

Revised: 27 December, 2023

Accepted: 27 December, 2023

Online: 20 January, 2024

### Keywords:

Semantic Resume Attributes

Comparative Description

Ranking

## ABSTRACT

A resume is a prevalent and generally employed method for individuals to showcase their proficiency and qualifications. It is typically composed using diverse customized, personalized methods in multiple inconsistent formats (such as pdf, txt, doc, etc.). Screening candidates based on the alignment of their resume with a set of job requirements is typically a laborious, challenging, time-intensive, and resource-intensive endeavor. This work is crucial for extracting pertinent information and valuable attributes that indicate acceptable applicants. This study aims to improve the candidate career-matching process for human resource (HR) departments by implementing automation and increasing efficiency. Machine learning (ML) and natural language processing (NLP) techniques are applied to infer and analyze comparative semantic resume attributes. Using semantic data comparisons, the ranking support vector machine (RankSVM) algorithm is subsequently applied to rank these resumes based on attributes. RankSVM detects tiny differences among candidates and assigns unique scores, resulting in an improved ranking of candidates based on their suitability for job requirements and from the best to worst match for the vacancy. The experiment and performance comparison results show that the proposed comparative ranking, which relies on semantic descriptions, outperforms the standard ranking based on regular scores in distinguishing candidates and distributing resumes across the ranks with an accuracy of up to 92%. Eventually, we obtained a list of the top ten candidates out of 228 technical specialists' resumes.

## 1 Introduction

This paper is an extension of work initially presented at the 2022 14th International Conference on Computational Intelligence and Communication Networks (CICN) [1], which improves the candidate-career matching process. In the beginning, the Internet has played a leading role in several interactive communication fields. It has become crucial for job seekers and employers to find distinct candidates for their vacancies. Nowadays, job applications are predominantly conducted via online job portals or services. The job requirements are published by a human resources (HR) officer, while candidates upload their resumes via a specialized recruitment website.

Subsequently, the process of selecting an appropriate candidate for the job requirements requires time and human effort. Recruiters search available resume databases using a job-related keyword to select pertinent resumes for a specific job, but the traditional engine search is based on keyword filtering techniques without understanding the related semantic information of different resumes. This can

lead to similar resumes, which may not be closely related [2].

For example, in the technical domain (computer-related technologies), special information and common resume features such as skills, tools, and technologies may exist in the same single domain as computer science. Therefore, the capability of extracting all of these special skills and attributes can distinguish each resume from the others resumes, which is usually performed manually and is preferable to be partially/fully automated because it is an important, complex, and time-consuming task within any HR [3].

A resume is a document, either in printed or electronic form, that presents job-related knowledge and information to employers. The resume includes personal information, education information, work experience, technical skills under hard skills, leadership, creativity, time management, qualifications, and preference sections under soft skills. Each is written openly based on the author's personality and writing style. Furthermore, a resume typically comprises a document in which the information is presented in various file formats and structures. Therefore, it is probable that the format and content of numerous resumes submitted by applicants vary, necessitating a

<sup>\*</sup>Corresponding Author: Asrar Alderham, King Abdulaziz University, KSA, +966559813248 & [aalderham@stu.kau.edu.sa](mailto:aalderham@stu.kau.edu.sa)

review, evaluation, and filtration process to ensure improved candidate selection. HR job sections may, therefore, encounter difficulties and challenges when attempting to extract useful information and select potential candidates.

It is, therefore, imperative to keep pace with the astounding advances in machine learning (ML) technologies, as computing power becomes challenging when job requirements require further analysis and re-filtering of resumes by setting more precise criteria for extracting relevant resumes.

Much research has been done in resume analysis, including classification, summarization, and extracting information using various techniques such as ML, natural language processing (NLP), and Ontology. Such research efforts have faced many problems regarding the accuracy of selecting the most eligible candidates' resumes. Extensive resume analyses and further research studies are still needed to address these problems.

In this work, we aim to assist HR officers in improving the extraction, filtering, and selection of the most job-suited resumes in an automated and more accurate manner based on analyzing semantic resume attributes using NLP tools. Our contribution considers two groups of per-attribute semantic descriptions used for each resume as relative and comparative labels. This comparative description is more precise and informative than categorical descriptions. However, comparative labels must be derived from all possible comparisons between every two resumes per attribute. Moreover, using a ranking method like the ranking support vector machine (RankSVM) algorithm [4], results in discriminative ranking for all resumes per single attribute, per multiple attributes, or by all attributes.

The rest of this paper is organized as follows. The following sections 2, 3, 4, and 5 presents background in brief, the process of extracting information from resumes and candidate-career matching, relative and comparative descriptions. In addition, the motivations, the research paper question, and the work objectives. A reference to related works on resume analysis that employ various methodologies and strategies is provided in section 6. Section 7 subsequently provides a comprehensive outline of the methodology employed in this study and the proposed approach. After that, in section 8, the dataset, the experiments, and the results are described in detail. The paper is concluded in the 9 section, and in the 10 section, we suggest directions for future work.

## 2 Background

A brief background on candidate-career matching, resume parsing, and the information extraction process will be given in this section. Then, the relative and comparative descriptions will be highlighted.

### 2.1 Candidate-Career Matching

Job/career matching is an operational issue that is ever more important in every society. A candidate-career (job-candidate) matching task aims to assign the right job to the right candidate (applicant). Candidate-career matching is a tedious process that HR officers carry out. There are so many resumes to analyze in a limited time, with a lot of work to pass through each candidate's application and identify the best matches for the job. Hence, unfortunately, any

possible negligence in this process may lead to the wrong candidate being put in the wrong job by a human mistake [5], which in turn can cause a loss for both employers and candidates.

There has been a rapid increase in jobs and employment in recent years through numerous internet recruitment platforms. Moreover, nowadays, because of the COVID-19 pandemic, millions of employers and job seekers would prefer or be obliged to conduct their hiring or job-seeking attempts besides job interviews through an online recruitment platform [6].

In [7], Russell stated, "Core personality is made up of traits that have been conditioned over many years. Such traits are critical in assessing a candidate's ability to perform virtually any aspect of any job". He also stated in the candidate's selection process, "The selection process is clearly the most critical and controllable variable in the development of a productive and successful work team."

The proper match between the candidate and the job is crucial for HR in hiring the right candidates, and it is also beneficial for those candidates to avoid involvement in work that is not right for them. As a result, increasing the matching process of candidates with occupations through proper resume selection and analysis is crucial for improving accurate applicant employability and developing a successful employment process, which in turn helps enhance employee job performance.

Three factors can affect a person's job performance capabilities as mentioned in [8], which can be valuable and insightful for this research work:

#### 1- Organization Match

The degree to which a candidate's attitudes, values, ethics, and grooming match those required by the job is called an organizational match. Face-to-face interviews are commonly used to assess these factors. However, relying solely on face-to-face interviews is risky. The halo effect is an issue that occurs when interviewers see a part of themselves in a job candidate. On the other hand, the opposite of the halo effect is another potential pitfall of the interview process: unconscious bias. The more a candidate differs from the interviewer, the more conscious effort the interviewer must make to view the candidate positively or neutrally.

#### 2- Skills Match

The degree to which a candidate's educational background, technical skills, previous job experience, and specific expertise match those required for the job is referred to as a skills match. Many job positions necessitate specific sets of knowledge or technical skills. According to research, those in charge of selecting these positions are frequently biased toward believing that expertise and intelligence are significant. However, more than expertise and skill intelligence are required. According to job matching research, people perform better when fully engaged in the challenges of the job.

#### 3- Job Match

Job match is a major component of a candidate's success on the job. It refers to how much a person's cognitive skills, such as how fast and effectively they learn. Interests include whether a person desires to work with people, data, or things and personality, such as the ability to be part of a team, make decisions, manage clients, etc.

All three factors or cornerstones of job performance are essential in matching the candidate to the job. Therefore, in this research, inspired by these pivotal factors, we focused on the applicant selection

process based on the job requirements by analyzing semantic attributes provided in their resumes and finding the differential aspects between them.

## 2.2 Resume Parsing and Information Extraction

Resume parsing is the automated extraction of information from websites or inconsistently formatted documents, such as resumes, using complex pattern matching or language analysis tools. This procedure aims to create a possible recruiting database in various formats [9].

Resumes can contain semi/unstructured text, vary in information type, order, and writing style, and represent various file formats (.txt, .pdf, .doc, .docx, .odt, .rtf, etc.). A resume is typically divided into sections reflecting the candidate's competencies. Due to this diversity, resumes are difficult to parse, and extracting useful information is challenging. The recruitment or HR staff spends significant time and effort parsing resumes and extracting relevant data [10]. So, it is crucial to have an accurate parsing system for the resume section. To effectively and efficiently analyze the data from various forms and structures of resumes, the automation model must not rely on orders or types of information [11].

Information extraction (IE) technologies, such as NLP, take natural language text as input and help analyze the text efficiently and effectively to discover valuable and relevant knowledge that can be used to produce usefully structured information.

Since resumes are written in human languages, the computer used to parse candidate resumes must be constantly trained and adapted to deal with human languages and different expressions in the writing of resumes. The capabilities of machine/deep learning techniques, considered among the most effective approaches under the artificial intelligence (AI) umbrella, allow a model to learn patterns in data without being programmed [12]. Furthermore, NLP can also be employed for such purposes and used to understand human language in resumes and extract useful information embedded in them.

NLP can be described as using computational methods to process free text in spoken or written form, which serves as a mode of communication commonly used by humans [13]. The main objective of NLP is to analyze unstructured text and represent its meaning using pipeline processing steps. These processes operate at two levels: the syntactic level, which deals with the structure and grammar of the text, and the semantic level, which focuses on the meaning and interpretation of the text. The syntactic level involves dividing the raw text of a document into sentences using a sentence segmenter. Each sentence or statement is then further divided into words and punctuation, known as tokens, using a tokenizer. Subsequently, every token is assigned part-of-speech tags (such as nouns, verbs, adjectives, adverbs, etc.), which will play a significant role in the named entity recognition (NER) process. This step identifies all occurrences of a specific entity type in the text. The final step involves using relation recognition to search for possible links between the different entities in the text at the semantic level, where each word is analyzed to determine the meaningful representation of the sentence [14]. Figure 1 shows the architecture for a simple information extraction system.

We have this background to use as a basis for our goal. To

improve the process of matching candidates to job requirements, we propose a new approach that differs from existing work on the topic of resume analysis, where many of these existing related works will be reviewed in detail in section 6. Unlike traditional approaches, our proposed approach aims to enhance resume analysis using comparative descriptions. This involves extracting relevant information from resumes using ML and NLP techniques. These comparative descriptions are expected to be more precise and informative in distinguishing between resumes based on their semantic attributes. Previous research in various domains [4, 15], and [16] has shown that such comparative descriptions are superior to other forms of description.

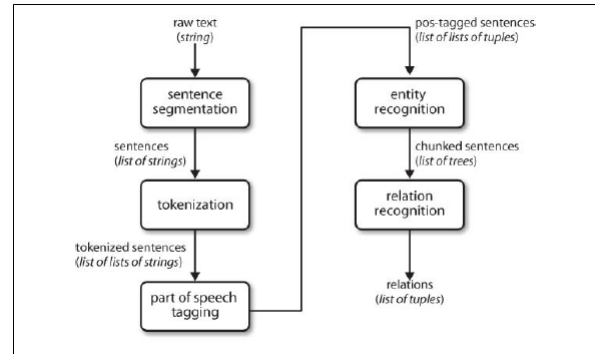


Figure 1: Simple pipeline architecture for an information extraction system [17]

## 2.3 Relative and Comparative Description

The parsing may fail and not extract high-quality data. If the search is more specific in describing job requirements, the search in most approaches is typically achieved based on an exact match of the search keyword or an explicit, predefined list of related words. However, other valuable implicit and semantic information is not analyzed or considered in the resulting retrieved data, which HR officers can usually infer and consider while manually searching and selecting candidates. Therefore, to fill this semantic gap between how humans and machines analyze resumes, we propose a new method to define several semantic attributes that describe different aspects of the process, like personal information, education level, experiences, certifications, etc. These semantic attributes may be explicitly or implicitly included in a resume. We need to parse them by searching keywords and giving each attribute a set of descriptive comparative labels demonstrating its significance.

Such semantic attributes can either be binary attributes associated with categorical labels or relative attributes associated with comparative labels [4]. Categorical (absolute) labels can be defined as nameable descriptions used to describe semantic attributes, such as education level, which can be labeled with any of the categories (BSc, MSc, Ph.D., etc.). Relative attributes are used to describe the degree of presence using labels such as "Very high," "High," "Average," "Low," "Very Low," and "None." Whereas comparative labels are named descriptions used to describe only relative attributes; for instance, they can be used to describe the education level of one resume based on the comparison with the education level in another resume. In other words, these labels describe the degree of comparison of relative attributes, using suitable labels such as

“Much higher,” “Higher,” “Same,” “Lower,” and “Much lower.” The proposed relative attributes are expressed using a bipolar five-point scale from 1 to 5. Corresponding comparative labels are assigned on a scale from 2 to -2. such that 2 is associated with the “Much higher” label, while -2 is associated with the “Much Lower” label. Thus, the relative attributes can represent the strength of the measured attributes but can also be comparable, making it easier to more accurately differentiate the minor underlying differences in the descriptive attributes between one resume and another [18]. As indicated in [19], there may be several expected advantages to describing the resume features in a relative (comparative) format:

1. It makes resume descriptions clearer and more meaningful (for example, resume A is better than resume B).
2. It enables comparisons with a reference resume object (e.g., resume A is higher than Asrar’s resume).
3. It makes attribute-based searching more efficient (e.g., search for a much higher resume with respect to skills).

### 3 Motivations

In our lives, getting a job that fits our ambition, qualifications, and abilities is very important, as matching our experience and knowledge with the job opportunities increases the efficiency of our production and progression in the labor market.

Since the advent of the Internet and the accelerating increase in online employment opportunities, matching candidates with job requirements is a tedious process that takes much time, effort, and cost to achieve. With many candidates seeking jobs, the need for a workforce increases to sort and analyze their resumes and choose the most suitable with high accuracy. For instance, when only two vacancies are required, and the applicants are more than a thousand candidates, each has his resume written differently to suit his thinking and how he presents his qualifications. It is often an inconsistently structured document that is difficult to analyze with the required accuracy automatically, ensuring distinguishing and the candidate matches the job from hundreds of others.

Proposing a comparative description-based approach adds to the existing resume analysis works a new viability expected to increase the current accuracy and efficiency of the selection of the candidates. It differs from merely comparing one resume with the other using traditional comparison methods. It compares the attributes of all resumes with each other based on a range that contains five points, making the selection and differentiation process for resumes more precise and informative.

### 4 Paper Research Question

Does the resume analysis based on semantic attributes described using comparative descriptions along with NLP and ML techniques can improve the process of candidate-career matching and selection?

### 5 Objectives

The main objectives of this work can be summarized as follows:

- To assist HR officers in improving the extraction, filtering, and selection of the most job-suited resumes in an automated, more accurate manner.
- To enhance existing work on resume analysis and candidate-career matching.
- To extract and organize the resume information related to a group of semantic attributes using relative /comparative semantic descriptions.
- To score the resume using initial relative labels and then compare them using comparative labels based on evaluation metrics for each semantic attribute.
- To rank resumes per attribute and then per all/some attributes based on comparative descriptions by matching the extracted semantic attributes from the resume to the job requirements.
- To improve the accuracy of candidate selection based on more accurate comparative descriptions, enabling the detection of tiny differences between candidates.

### 6 Related Work

Much work has been published on automatically extracting and analyzing information from resumes to improve candidate-career matching and selection. While many techniques and approaches are used in unstructured resume analysis, most researchers focus on ML and NLP to extract entities from resumes to ensure the best match between applicants and job requirements.

#### 6.1 Resume Analysis and Candidate Selection using NLP

Regarding resume analysis and the selection of candidates, in addition to the job domain allocation process and the classification of applicants in the field of ML and NLP techniques, several research studies have been summarized from 2018 to 2022 since they were interested in the current research scope, as will be illustrated about the studies [10], [20]–[43]. Started with [20], targeted at the challenge of resume data extraction using supervised and rule-based methods, which rely on hierarchical knowledge and large amounts of labeled data that are overly difficult to collect. They proposed a two-step of resume information extraction method. First, lines of raw resume text are segmented using a Naive Bayes (NB) classifier, which predicts each phase label in the resume. Second, after reducing text noise, term frequency-inverse document frequency (TF-IDF) cosine similarity is computed, and K-means clustering shows the resume attribute cluster. Writing style is a new feature proposed, whereas each line’s writing style comprises a word index, punctuation index, lexical attribute, and classifier prediction results. So, in the second step, the writing style is utilized to determine each semi-structured text relevant block and module components to get the structured resume text. The technique is practical and efficient, as demonstrated by experiments on a real-world dataset on information technology (IT) professional resumes.

To create a model that could extract valuable information from resumes, the authors of [21] used both Rule-based and Deep-learning approaches. Their work focuses on Vietnamese resume files in the IT domain. The approach of this study contains 4 phases: First, text segmentation; second, named entity recognition (NER); third, a combination of convolutional neural networks (CNN), bidirectional long short-term memory (Bi-LSTM), and conditional random field (CRF) to find name entities, and finally, text normalization. Above 81% accuracy in NER is a significant achievement for their model. This work demands a lot of time and people if they want an extensive enough data set.

In [22], the authors explained the proposed system used, which improves the decision-making process regarding candidates, as it consists of multiple modules: the section-based segmentation module, the filtration module, and the category-based matching module. They deal with the problem of selecting candidates who match the job requirements, and the human resources staff also exerts manual effort. They also addressed the issue of extracting information from unstructured resumes. The system uses an ML algorithm to build the applicant learning model and NLP to extract information. Note that this work covers resume analysis with all its sections and information like personal, education, experience, technical skills, internship, hobbies, etc. However, a limitation of the system is the absence of a research sample to evaluate the accuracy and efficacy.

In [23], they addressed two issues; the most significant was obtaining employer-interesting information from resume content. The second issue is structuring resumes automatically for database entry. They constructed a computer software that answers any question about resume content. A question is posed as a string of keywords that may not need to be grammatically correct. The response is supplied as a text block or a single resume clause. Their process of resume-text processing has four steps. Convert a resume from Word or PDF to text by Tika system, identify text keywords, create two directories, each with a  $\langle$ key, value $\rangle$  pairs, and finally, text diapason definition. This paper gives examples of resume text extraction to answer arbitrary queries. Their approach allows for a structured resume through clustering and keyword labeling. The shortcoming of these methods is that they apply to process only any short text.

The study in [24] developed a web application that predicts the best resumes for a particular job description. The web app's design has two types of users: candidate and recruiter users. On the candidate side, the SpaCy model and regular expression (RegEx) of NLP extract different sections of data from the uploaded resume. On the recruiter side, Job descriptions, like resumes, must be scanned and parsed to extract essential data. Then, the TF-IDF cosine similarity compares a candidate's resume to the job description. Finally, the output is a ranked list of applicants based on total resume scores visible only to the recruiter. There were problems with parsing unstructured resumes; the candidate's education years were added to their job experience years, which were calculated. So, the applicant had to provide manual input for their job experience to count toward their overall resume score, in addition to the web application problems in working on .doc files only.

To avoid the time-consuming and redundant process of manually allocating projects to new hires by opening and analyzing their resumes one by one. In [25], they discussed developing and implementing a resume classifying application that uses NLP to

obtain only the essential and relevant information. In addition, an ensemble learning-based voting classifier consisting of 6 individual classifiers to classify a candidate profile into a suitable domain based on his interest, work experience, and expertise mentioned in the profile. The resume classifier application produces a bar graph showing candidates' suitability for various domains. The model uses Stack-Overflow REST API topic modeling techniques to add a new domain to the list of domains if the candidate profile fails to meet the confidence threshold value. The model results are encouraging, and their studies are distinguished by their focus on IT.

In [26], the authors offered a two-step process to rank applicants according to how well they match the requirements of a job vacancy. The first step is to create a resume parser that extracts complete information. This parser is a web app. In the second step, they employ bidirectional encoder representations from transformers (BERT) as sentence pair classification to rank job candidates based on matching the job descriptions by using a candidate's past resume experience descriptions. Their experiments first extracted LinkedIn resume text into predefined sections and achieved 100%

In [27], they created a resume analysis and position recommendation system that included competitiveness, personality traits, and job recommendation analysis with resume diagnosis. The diagnosis function uses ML and text mining to analyze the user's uploaded resume. It helps job applicants understand their competitiveness in the market and generates a list of recommended job openings based on their resumes. Job-seekers and companies immediately summarize their analysis results in reports and charts for job fairs. The job vacancy recommendation results were evaluated using a questionnaire survey on the applicant side. The experimental results confirmed that the job vacancies recommended by the created system met the expectations of job candidates.

One research work using NLP techniques for extracting information from legal documents is [28]. They have suggested using open information extraction from legal documents. Open information extraction involves extracting data from large datasets without a predefined data set. A sample case from the indiankanoon.com website is put into the system as a text document. As a first step, they performed data cleaning, including removing ambiguities, tagging different parts of speech (POS), chunking names, recognizing them using NER, and extracting relationships between them. No numerical results demonstrate the efficiency and accuracy of their proposed approach.

As the traditional recruitment process lacks speed and accuracy in selecting the best match of candidates for the job, a new model was proposed by [29]. They used the NLP method to summarize resumes in different formats by extracting important information for the job requirements focused on the technical skills and listing each summarized resume in a .txt file. The summarization model specification consists of four integrated modules implemented in Python: conversion, extraction, general, and resume list. The research concluded that the auto summarization model of resumes is not an efficient solution because it might not work well in all industries regarding the different skill sets needed in the different areas of recruitment.

In [30], they suggested an automated method for 'resume categorization and matching' to speed up the candidate selection and

decision-making process and further overcome obstacles to the proper screening of candidates. Their model is built on ML, which makes resume recommendations to HR based on job descriptions. This model works in two phases: extracting the features using NER of NLP and then classifying the resumes into the correct categories using LinearSVM—moreover, finding similarities between the resume and the job description using TF-IDF cosine similarity. The authors demonstrated the result of their model in the resume classification phase, with an accuracy of 78.53%.

Traditional hiring requires manual resume screening, which takes time and energy. Advanced machine learning-based NLP automates the resume screening and hiring process in this paper [31] by extracting entities from resumes using SpaCy and NER models and generating a graph showing each resume's score. The model is evaluated on 20 resumes, and the predicted summary resumes are saved as .txt files for each. The experiment achieved an F1-score of 91.35, a precision of 90.14, and a recall of 92.60. Their entity-recognized model achieved a 91.35 F1 score, 90.14 precision, and 92.60 recall.

In [32], they proposed a resume parser system using NER of Stanford CoreNLP and pattern matching to convert the resume data into a structured format and extract information about the candidates for the recruiting process. Based on skills extracted using TF-IDF and logistic regression classification techniques, the system predicted the candidate's genre, such as computer science, accounting, finance, statistics, business development, etc. The experiment was conducted with 100 resumes from students and job seekers from various universities. The result of the overall resume prediction accuracy of their system was 91.47%.

In this proposed system of [33], the authors made the entire enlistment process more practical and economical. As a result, they have put in place a system that compares applicants' resumes with job descriptions and displays the similarity percentage. Cosine similarity is the basis of the system under consideration. The NLP technology extracts the knowledge and vital abilities required for the specific employment position. It shows the recruiter the candidates' similarity results, which aids in selecting the best candidates. Additionally, candidates are ranked based on their similarity score to obtain the most cost-effective outcome possible. HR departments would reduce their burden if the system were implemented across various industries with a high demand for qualified workers.

Because resumes are unstructured, human-written documents, it is urgent to understand the context of the words. So, in [10], they addressed the problem of the time and effort expended to analyze and extract information, especially the parser of special skills from resumes. They proposed semantical and contextual rich IE using the advanced NLP library SpaCy, which has a feature called "Phrase Matcher". Their approach identifies a table or dictionary with the various skill sets, parses the resume to search all skill sets, and counts the frequency of those words of different categories. To select the appropriate candidates, they used the Matplotlib tool to represent the information visually after parsing. From 250 resumes and job requirements specified in the dictionary used in the result, they found two candidates' resumes satisfying the job requirements. Note that their research was limited to extracting special skills from the resume, which is only one of several other important parts such as experience, education, etc. Therefore, selecting candidates is

done visually, not using ranking algorithms.

In [34], they developed a job portal where employees and applicants can post their resumes, making recruiting easy and efficient. Their portal aids in the organization of resumes according to the needs of specific employers. Also, they used data from social media like LinkedIn to help recruit high-quality candidates worldwide while avoiding unfair and discriminatory practices. The result involves converting resumes from /pdf/doc/RTF to plain documents and tokenizing data entities using optical character recognition (OCR), NLP, and ranking algorithm tools. When comparing extracted entities and needed keywords, resumes were ranked based on their technical skills, and the results were supplied in the form of a pie chart and bar graph.

The research work of [35] was to develop a system that automates the eligibility examination and evaluation of candidates in recruiting students for job vacancies or higher education programs. This system handles the manual analysis of resumes with all its tedious tasks and provides accurate and practical evaluation results. They used ML approaches, NLP, and three classification algorithms to implement the system. The evaluation of this work was divided into several steps conducted on personal information, professional experience, academic background, and soft and technical skills. The results obtained in the classification are cross-validated by the results of online video interviews. This work used acceptance criteria for candidates based on the scores given after extracting the attributes, which, as a result, is an indicator in which resumes are classified between acceptable and rejected. Hence, this is what mainly differs from our proposed approach, which systematically ranks the candidates starting with the most matching the job requirements based on relative and comparative scoring instead of merely judging them with binary labels like (acceptable or rejected).

In [36], the authors address the problem of resume analysis and the complex use of the parser when supporting languages other than English. They used ML techniques in the context of NLP that achieved high accuracy in extracting information from resumes in an arbitrary format and five different languages. They created a system with many interconnected models, using state-of-the-art NLP models as a basis. Their approach used a new deep model architecture for sequential input data, called a transformer, with parallel processing of the input sequence in the form of the BERT language model. BERT is designed to learn deep bidirectional representations on unnamed text. Their models extract and categorize relevant resume sections (personal information, skills, education, previous job) and corresponding specific information at the lower hierarchical level (names, dates, addresses, competencies, etc.). Models were evaluated on a data set of 1,686 resumes. For Norwegian, Swedish, Finnish, Polish, and English languages, the system achieves F1-scores of 0.86, 0.88, 0.86, 0.87, and 0.82 at the section level, respectively, and 0.75, 0.80, 0.80, 0.81, and 0.83 at the item level, respectively.

One of the essential parts of a resume is the educational qualifications section, which captures the knowledge and skills relevant to the job. In [11], they solved the problem of a large amount of annotated data required to determine educational institutions' names and grades from a resume's education section. They proposed a semi-supervised model for accurately identifying degrees and insti-

tute names on a resume based on NER. It consists of CNN, word embeddings, and BI-LSTM. This model is used to predict unclassified education section entities and is corrected using a correction unit. This model was evaluated on 500 training and 50 test resumes, achieving an overall F1-score of 73.28 and an accuracy of 92.06%

Higher-level skills can be deduced from lower-level ones, and vice versa, rather than simply pulling out terms associated with competence. Extracting skills is a vital step in developing job recommendation systems. In [37], they proposed using CNN to create an explainable model that can extract high-level skills from resumes in their raw text format. The resulting model can predict the high-level skills mentioned in a resume and highlight the underlying low-level skills that led to that prediction. Experiments were conducted on anonymous IT resumes collected from various websites. The overall model achieves 98.79% recall and 91.34%, and more than 99% accuracy for specific skills.

In [38], they have proposed a two-stage embedding-based recommender system for matching available jobs with suitable candidates. A component for candidate retrieval using fused embedding and a module for fine-tuning and reranking candidates. Deep learning using CNN, representation learning, job-skill information graph, and geolocation calculator are fused for the job and candidate embedding. They have also implemented the Faiss index for clustering and compressing embeddings, which allows conducting runtime nearest neighbor searches. The final ranking score is calculated by a weighted linear equation that aggregates the first-stage relevancy score and contextual job and candidate features. Their job-to-candidate matching system has a satisfying title, description, requirement, and location-matching quality results. Their system has vastly improved the matching quality between jobs and applicants.

Discovering top candidates with few resources in a short time is the most pressing problem facing businesses today. In [39], the best candidates could be ranked using content-based suggestion, which employs NLP techniques to parse resumes. With an average parsing accuracy of 85% and a scoring accuracy of 92%, the system performs effectively. In addition, cosine similarity is used to discover the resumes most relevant to the job description provided, and the k-nearest neighbor (KNN) algorithm is used to select and rank Job applications.

People may have a better chance of finding jobs that will allow them to have a good life if they have the education level appropriate for the professional environment. In support of the impact of education on job applicants, the authors of [40] addressed gaps between the job market, job seeker, and educational institution skills. Recruiters want to check resumes for required skills automatically. Job applicants want to know what skills must be added to their resumes. Educational institutions recommend study programs and assist students in ensuring their courses cover job posting skills. Since all three of these users have skills in common, they made an app called "Skill Scanner," which uses NLP techniques to analyze, vectorize, cluster, and compare skills. It then makes reports with statistics and suggestions for all three users of which skills are covered and missing. They analyzed the master's program in data science and data scientist positions from Indeed.com and Kaggle.com. As part of a questionnaire, 108 representatives from their three parties were provided reports generated by Skill Scanner. Most users report that

their solution improves the efficiency, speed, fairness, explainability, autonomy, and support of skill-related processes. They proved that 89% of those who tried out their recommendation system had no problem using it.

Categorizing job applications received as resumes against open vacancies requires a significant amount of time and effort from an employer. This study [41] aims to develop an automated resume classification system (RCS) that classifies resumes according to their job categories. This study's main contribution is in preprocessing the resumes to a corpus and extracting vectorized representation using NLP techniques appropriate for classification tasks performed by ML algorithms. This work was evaluated on three extraction and representation techniques and nine ML classification models. The TF-IDF vectorizer was the best at extracting features and representation, and the SVM classes performed exceptionally well on parsed resume datasets with more than 96% accuracy.

In [42], authors introduce I-Recruiter, an intelligent decision support system (IDSS) for screening resumes and identifying the most qualified applicants for available IT sector vacancies. The semantic similarity between a resume and the job description determines an applicant's ranking in I-Recruiter. It provides information about the best applicants so that the hiring process can move forward. ML and NLP constitute the core of the system's functionalities. The essential components of this system are the training, matching, and extracting phases. The training block produces domain-specific word embeddings that have been trained. At the same time, the matching section identifies top candidates by comparing resumes and job postings for semantic similarity. In the last phase, It extracted some primary data on the top applicants. I-Recruiter showed excellent results, with an average accuracy of 96% and a short amount of working time.

Recently, In [43], they use NLP to extract useful information from resumes. The system will then go to the person's profiles on LinkedIn and GitHub, scrape information from those sites, and feed it into the ML Models to make a more accurate prediction. The first ML model, either KNN or SVM, predicts what kind of job role their resumes are best suited for it. The second model recommends improving their resumes by using cosine similarity, which compares the user's input to the model's prediction. They proved that the accuracy of the models ranges from 78% to 98% depending on the datasets utilized, the learning methods' complexity, and the dataset's size.

Several studies like [44, 45, 46, 47, 48, 49], and [50] suggest using ontology-based knowledge for resume parsing, an ontology is a knowledge base representation that can generate a semantic model of data connected with specific domain knowledge. Furthermore, ontology is utilized to define linkages between various types of semantic knowledge in a domain. Semantic web models use ontology to define the meaning of target data, information, or knowledge. The most fundamental comprehension of the semantic web is surpassed by ontology, which provides the ability for standard reasoning, typically based on the specification of inference rules. Ontologies let users organize information in taxonomies of concepts and their attributes and describe how these concepts relate to each other [51]. The authors of [52] say that there are two main reasons to use ontologies: to help people and software agents understand how information is structured and to make it possible to reuse existing

domain knowledge.

In [44], a resume ontology based on skills was created, which refers to the most necessary parts of a job to see how well the applicant matches the job description. Applicant and job profiles were annotated with a common vocabulary. The semantic concept similarity algorithm was modified to accurately compute and rank matching scores between profiles when a query was run. Based on the results of the system's experiments, the approach improved the search matching accuracy.

The author in [45] built an ontology that contains the most important factors to consider while hiring for IT positions. The primary purpose of the proposed approach is to reduce errors in the first stage of personnel selection by manually filtering hundreds of resumes/profiles to choose applicants for interviews. Their suggested system consists of a mobile application that automatically chooses online profiles from professional websites (like Indeed, LinkedIn, and Monster) and ranks them to eventually display the appropriate candidates for a given vacancy to the recruiter. They created an ontology focused on resume skills to match candidate resumes to job description requirements. GATE and Apache Tika tools were used to automatically extract skills from unstructured resumes in the implementation's first step. The specified information will be used to construct a structured RDF document, saved in a triple store, and queried for each job offer.

A supplement work that used the ontology for resume analysis is presented in the proposed research of [46]. The authors used text mining and ML tools to make an effective company recommender system (CRS) that could help recruiters find the best candidate for a given job title. Because the data to be extracted is unstructured, the authors used traditional and ontology-based information extraction strategies. Candidates for IT organizations were classified into three categories: low, average, and high, based on their rating score, which was interpreted as their competency level in the programming language mentioned in their project description, according to the model proposed. Finally, the organization can use the ranking to select the best possible candidates for the job openings.

The exploration in [47] also sought to address the issue of analyzing unstructured resumes. In addition to the limited availability of language dictionaries and the complexity of Polish linguistic dependencies, this drives their interest. The proposed prototype designed a hybrid resume parser service using text mining algorithms to extract information from Polish resume documents for IT recruitment. The resume parsing system combines three NER tools (Liner2, Nerf, Babelfy) with the anchor NER service and dictionary methods such as Fuzzy Dict and Competence Lexem. The proposed system also included a promising CRF method. Data from an IT recruiting firm was used to conduct the research, and the results showed that the hybrid approach proposed improves single results obtained for detecting educational institutions. The result of the combined tools (hybrid approach) was more than 60% better than the best single solution (30.92% vs. 19.23% for Babelfy).

The authors in [48] have considered the effects of ontology-based use in the resume search system for job applicants. They provided an ontology structure for representing resume and job description (JD) contents, extracting information from collected resumes and

JD using OCR, and classifying extracted information using an unsupervised approach called computer science ontology (CSO) classifier. This ontology searches resumes for job skills that match JD criteria. The challenge of candidate ranking in an automated recruitment system is solved using ML and NLP. This solution is based on an IT ontology called Job-Onto. Moreover, the planned system is implemented as a recruiting website that helps candidates and employers find opportunities and streamline their recruitment processes. The suggested recruitment system uses automated resume ranking based on criteria, with the recruiter controlling the weighting. Each time a job is posted, the system calculates the resume correlation score for each candidate. This score is the sum of their domain skill, general skill, and soft skill matching scores. Finally, a list of the best resumes for the front-end engineer job is shown. The proposed method proved effective in finding resumes that correspond with a particular JD in the IT industry, according to test results.

In [49], the authors proposed an approach to improve selecting the best candidates, including three primary processing steps: text extraction, text block classification, and resume facts identification via NER. A custom-built ontology is used to augment the extraction of technical skills. This research is unique because it focuses on many areas, including diversity and improving electronic recruitment re-usability. However, there are no numerical findings for the accuracy they intended to improve in extracting information from personal resumes.

Every job has distinct requirements: some require the most experience, education, or skills. Finding the proper applicants in a large resume pool is difficult, especially when we are looking for specific skills. So, in [50], their goal is to create a system to analyze resumes and job postings based on how well they match one another. Then, rank them in ascending order of score. They used ruled-based methods, developed their own ontology to facilitate matching and scoring, and employed NLP methods for information extraction. Their ontology contains 37493 distinct nodes, and 54632 relations were created. They did real case studies to evaluate the AI scoring and other features. For actual job vacancies, recruiters and the system simultaneously searched for candidates. The system's top candidates were ranked.

Finally, we highlighted a study that focused on sentiment analysis regarding personal interviews with candidates and used the SVM ranking algorithm that we will apply in our research. All users on social media can write their sentiments to express their emotions and opinions. Three processes comprise sentiment analysis: sentiment extraction, keyword preparation, review analysis, and classification. Moreover, Covid-19 has shifted all processes online, such as video conferencing. This study in [53] provides a job candidate rank (JCR) model for job candidates based on the interviewer's sentiment analysis. The candidate rank model has stages—first, sentiment text parsing, stopword removal, and stemming. Second, select features using document frequency (DF), expand using the WordNet database, and focus groups on ranking them. SVM and NB of ML classifications are included. Using the NB classifier, the model achieved 93% accuracy, while the SVM classifier achieved 89% accuracy.



## 6.2 Comparative Description Approach

This research was motivated by the successful use of comparative descriptions in other applications; therefore, we aim to apply and extend such effective capability as a novel contribution to resume ranking and selection processes for effective candidate-career matching. The ability of comparative descriptions regarding resume analysis has yet to be extensively investigated. Thus, we widen the exploration to embrace other prominent applications utilizing comparative descriptions, such as human identification by semantic/soft biometrics.

One of the challenges of human identification and semantic characterization is the quality of soft biometric description [19]. A search in soft biometrics showed better accuracy of comparative traits or attributes than the accuracy of categorical traits [4, 15, 16, 19].

In [16], the authors compared subject comparisons in the Soton gait database by comparing one target to multiple subjects. When comparing one target to more than one subject, they found that the comparative descriptions performed 17% better than the absolute/categorical descriptions.

Also, in [4], the authors used semantic clothing traits as soft biometrics for human identification. They further explored their validity and efficiency through corresponding comparative descriptions, allowing for more accurate differentiation.

The study in [15] analyzed a data set using gender as a comparative attribute and found that comparative annotations are more discriminatory than categorical labels. The study's approach on 100 annotated subject images showed correct-match reliability in the top 7% with ten comparisons or the top 13% with only five comparisons.

Additionally, in [19], the authors studied human identity through comparative facial soft biometrics using the labeled faces in the wild (LFW) dataset. Comparative soft biometrics allows each person to be uniquely identified in the database by creating a biosignature with their exact physical traits compared to others. Such comparative descriptions improved searches based on a given comparative trait (e.g., searching for someone younger).

The last study [54] focused on enhanced human-machine communication and demonstrated the benefits of relative attributes on four applications. The applications include active learning of discriminative classifiers, zero-shot learning from relative comparisons, automatic image description, and image search with interactive feedback. Relative attributes compare an image's attribute strength to others rather than predicting its presence. Relative attributes would be more natural and allow for richer communication, more detailed human supervision, higher recognition accuracy, and more informative descriptions of new images. The SVM ranking function is learned and used for each attribute to predict the attribute strength. They demonstrate that relative attributes lead to higher performance across all applications by comparing them to numerous strong baselines using image datasets of scenes and faces.

All previous research works aim to bridge the semantic gap between biometrics and human description by using relative and comparative descriptions of attributes. In particular, to the best of our knowledge, no study has used comparative descriptions for semantic attribute analysis of resume documents for improving job-candidate matching.

Table 1 presents our proposed approach compared with several previous studies.

## 7 Methodology

As in the literature, there is a need to improve the process of automatically extracting data from resumes and mining their information [55]. The improvement can assist in selecting candidates and ranking them according to the job's requirements. Recruiters may have subjective ideas about which aspects they want to emphasize, depending on the circumstances and nature of their company's requirements at that particular time. Therefore, in this research, we proposed a novel approach to improve the candidate-career matching process based on comparative semantic resume analysis. Our approach comprises three phases: the semantic attribute extraction phase using NLP techniques, the relative and comparative labeling phase, and the ranking phase using the ranking SVM algorithm. The proposed model can generally analyze the resume by finding the exact keywords for each resume section, which will be used to assign a suitable label for 13 semantic attributes each. Thus, our proposed model lets the recruiters modify the attribute-based searching as they want to depend on their own requirements. After constructing a table or dictionary, which covers various semantic sets of attributes, count the occurrences of the words belonging to different semantic attributes and assign scores for each attribute based on evaluation metrics, which are dynamic to change and determined by recruitment staff based on the job requirements. Then, aggregate all points for each attribute as regular scores and show the related normalization score. After that, we come to the comparative description part, to which we contribute to enhancing the resume analysis domain by inferring a suited relative descriptive label per attribute for each resume based on evaluating their given scores. Then, inferring corresponding comparative descriptive labels by describing a pairwise comparison per attribute for all of the attributes between each resume and all other resumes in the dataset. Finally, ranking all resumes per attribute using the Rank SVM algorithm through extracting usable relative measurements from those comparative descriptions as resume scoring per attribute and, consequently, leading to overall resume scoring concerning all sets or a selected subset of attributes in descending order. This method helps HR query the list of candidates ranked based on each semantic attribute in the resume or multiple semantic attributes.

Our study aims to provide an ML-based model that does not solely rely on training data to match candidate resumes to career requirements. However, it also enables dynamic learning and ranking per any combination of the proposed resume semantic attributes. We summarize all of the methodology phases and present the flowchart of the proposed approach of semantic resume analysis in the next subsection.

Table 1: Comparing our proposed approach with previous studies

Study	ML	NLP	Ontology	Scoring	Ranking	Comparative	Language
[10]	√	√	-	√	-	-	English
[11]	√	√	-	-	-	-	English
[20]	-	√	-	-	-	-	English
[21]	√	√	-	-	-	-	Vietnamese
[22]	√	√	-	√	√	-	English
[23]	√	√	-	-	-	-	English
[24]	√	√	-	√	√	-	English
[25]	√	√	-	-	-	-	English
[26]	√	√	-	√	√	-	English
[27]	√	-	-	-	-	-	English
[28]	√	√	-	-	-	-	English
[29]	√	√	-	-	-	-	English
[30]	√	√	-	√	√	-	English
[31]	√	√	-	√	√	-	English
[32]	√	√	-	-	-	-	English
[33]	√	√	-	√	√	-	English
[34]	√	√	-	√	√	-	English
[35]	√	√	-	√	-	-	English
[36]	√	√	-	-	-	-	English, Norwegian, Swedish, Finnish, Polish
[37]	√	√	-	-	-	-	English
[38]	√	√	-	√	√	-	English
[39]	√	√	-	√	√	-	English
[40]	√	√	-	-	-	-	English
[41]	√	√	-	-	-	-	English
[42]	√	√	-	√	√	-	English
[43]	√	√	-	√	√	-	English
[44]	√	-	√	√	√	-	English
[45]	√	-	√	√	√	-	English
[46]	√	√	√	√	√	-	English
[47]	√	√	√	-	-	-	Polish
[48]	√	√	√	√	√	-	English
[49]	√	√	√	√	√	-	English
[50]	√	√	√	√	√	-	English
[53]	√	√	-	√	√	-	English
<b>Our work</b>	√	√	-	√	√	√	<b>English</b>

### 7.1 Summary of Proposed Approach

Our approach is proposed to assist HR officers in improving the extraction, filtering, and selection of the most job-suited resumes in the technical domain (or specialties) in an automated and more accurate manner based on analyzing semantic resume attributes. For each resume, two groups of per-attribute semantic descriptions are used as relative and comparative labels, which are more precise and informative than categorical descriptions. The following scenario will be followed in this research study:

1. Construct a table or dictionary that covers various semantic sets of attributes with each keyword in the resume, such as (personal information, education level, technical skills level, professional experience, personal skills, additional qualification, etc.).
2. An NLP-based tool is used to parse and extract information from the entire resume to search the words in the table or dictionary.
3. Count the occurrences of the words belonging to different semantic attributes and assign scores for each attribute based on evaluation metrics determined by recruitment staff based on the job requirements.

4. Aggregates all points for each attribute as regular scores and shows the related normalization score.
5. Inferring a suited relative descriptive label per attribute for each resume based on evaluating their given scores.
6. Inferring corresponding comparative descriptive labels by describing a pairwise comparison per attribute for all of the attributes between each resume and all other resumes in the dataset.
7. Ranking all resumes per attribute using the SVM ranking algorithm through extracting usable relative measurements from those comparative descriptions as resume scoring per attribute and, consequently, leading to overall resume scoring concerning all sets or a selected subset of attributes in descending order.
8. Analyzing results based on four comparison aspects which compare the efficiency in distribution and discrimination of resumes rankings.

Figure 2 overviews the proposed semantic resume analysis approach, and we will explain each phase in detail in the following subsections.

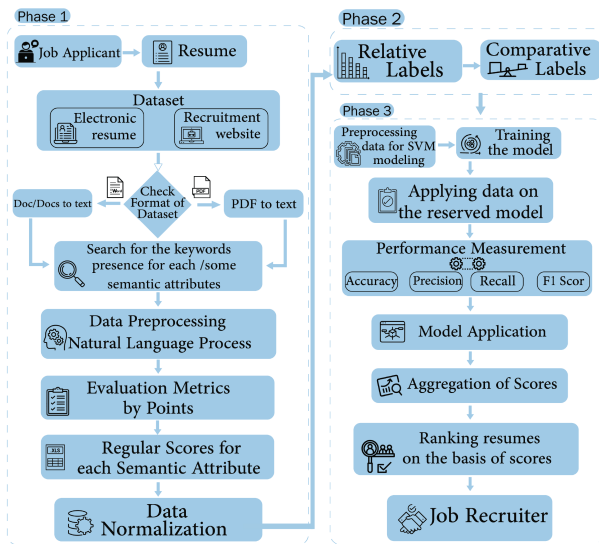


Figure 2: Overview of the proposed semantic resume analysis approach

## 7.2 Semantic Information Extraction Phase

An automated system for extracting information and selecting the potential candidates who best meet the position's requirements can significantly improve the efficiency of HR agencies [56]. However, resumes are unstructured documents with different file formats (pdf, txt, doc, etc.) and contain ambiguous and variable language. This heterogeneity makes it challenging to extract useful information.

To make the recruitment process simple, efficient, and automated, some advanced studies, like [10] and [33], solved the parsing problem with specific searches and unstructured data. They searched the whole document for words in predefined tables, dictionaries, or a BoW to list important words and count the number of detected words of interest. Therefore, we proposed a candidate selection system based on searching the keywords. If the keywords exist, it will count their presence and allocate the given points as evaluation metrics or as HR-specified job requirements.

Before we move to define the proposed semantic attributes, we will overview the NLP techniques in general and the related techniques we used in our approach:

### 7.2.1 NLP with Python

By "natural language," we refer to a human language used for everyday conversation, such as English, Arabic, Hindi, and Portuguese. In contrast to programming and formal languages such as computer and mathematical notations, natural languages have evolved as they are passed from generation to generation and are challenging to define with specific rules. NLP aims to "understand" complete human sentences so that appropriate replies can be provided. More and more technologies are appearing based mainly on natural language processing. In today's increasingly globalized and interconnected information world, language processing has assumed critical importance [17].

### 7.2.2 Proposed Semantic Attributes

Table 2 demonstrates the proposed semantic attributes that will be extracted from resumes and all relevant keywords and suggested evaluation matrices that will be defined and used in our approach.

### 7.2.3 Keywords Detection or Information Extraction

- Extracting files from the archive directory of resumes dataset and generating a list using `extractall()` python function.
- Checking the format of the input file from the dataset (pdf, doc/docx) and converting them to text using `str()` python function.

- Extracting semantic attribute information. We construct three functions (a function to check keywords, a function to check keywords and count their presence, and a function to check how many keywords are present). First, the personal information, name, email, and mobile number are parsed separately. Then, allocate the weights that the requirements illustrated using a simple resume parser existing in [57], which is used for extracting information from resumes by using the NLTK and SpaCy models of NLP. Since resumes have rather varied structures and formats, the keywords are similarly searched for the ten remaining semantic attributes. If the keywords exist, then the given points of evaluation metrics would be allocated. Aggregating all points for each attribute as regular scores and showing the related normalization score.

### 7.2.4 Data Normalization

For many procedures to work properly, the data must be normalized or standardized; since resumes have unbalanced scores after extracting attributes, we used normalization (MinMaxScaler) to transform features (the regular scores of the attributes) by scaling each attribute to a corresponding range (between zero and one). Then, we used the normalization score to get the relative scores, which range from 0 to 5, as explained in the next section.

The transformation imported using Scikit-learn of ML library is given by:

$$X\_std = \frac{(X - X_{\min}(axis=0))}{(X_{\max}(axis=0) - X_{\min}(axis=0))} \quad (1)$$

$$X\_scaled = X\_std * (max - min) + min$$

where  $min, max = feature\_range$ .

Table 3 shows an example of evaluation metrics by scores given for the personal information attribute, where the total is eight points for the whole personal information attribute and shows the related normalization score, which is calculated by "(1)" where  $feature\_range=(0,8)$ ,  $X=8$ ,  $min=1$ , and  $max=8$ .

Table 2: Proposed resume semantic attributes to be extracted and analyzed

Semantic attributes	Keywords
1- Personal information completeness	Name, Nationality, Mobile No., Email, Address, Phone, Contact No, Country, ...etc.
2- Education level	GPA, Bachelor, Master, PhD.
3- Technical skills level	Programming Languages, Tools, Framework, Databases, Operating systems, ...etc.
4- Professional experience	Experience, Participated, Worked, ...etc.
5- Personal skills (Soft skills)	Communication, Teamwork, Time Management, Problem-Solving, Creativity, Leadership, ...etc.
6- Awards	Awarded, Winner, ...etc.
7-Hobbies and interest	Play, Games, Surf, News, Design, Read, Video, Sports, ...etc.
8- Additional Qualification	Course, Program, Trained, Level, ...etc.
9- Professional Certificate	Certified, Cisco, Scrum, Professional, Management, Administrator, ...etc.
10- Suitable age requirement	Age, Years old, Birth, Birth year.
11- Career objective	Gain, Objective, Opportunity, Looking for, Looking forward, ...etc.
12- Project Experience	Project, Role, Client, Responsibilities, Environment.
13- Languages	Arabic, English, ILETS, STEP, ...etc.

Table 3: Example of evaluation metrics by scores, regular score, and normalization score for the personal information attribute

Personal information	Score
Name	2
Nationality	1
Address	1
Mobile No.	2
Email	2
Total (Regular score)	8
Normalized score	1

### 7.3 Relative and Comparative Labeling Phase

After we extract the important semantic attributes from the resume, we come to the comparative description part, to which we contribute to improving the resume analysis domain. In our work, apart from simple traditional evaluation metrics (a regular scoring system), we use the proposed comparative and relative labels to score or rate a resume more precisely.

After normalization, the points are totaled for each attribute in the regular scoring system to infer the relative scores. We multiply each point by five and then round it down. The labels of relative descriptions are defined as "Very high," "High," "Average," "Low," "Very low," and "None." Table 4 shows an example of relative labels and their corresponding scores.

Table 4: Relative labels and corresponding scores

Relative score system	Relative labels
5	Very high
4	High
3	Average
2	Low
1	Very low
0	None

After assigning the most appropriate scores based on relative labels for each semantic attribute in the resume, we will accordingly

use comparative labels to describe a pairwise comparison per attribute for all of the attributes between each resume and all other resumes in the dataset (one subject with all others). The comparative labels include "Much higher," "Higher," "Same," "Lower," and "Much lower." For example, when the relative score of personal information is 5 for resume A and 4 for resume B, then the comparative label for resume A will be labeled as "Higher" than resume B, with the new corresponding score equal to 1, as listed in Table 5, which shows an example of comparative labels and their corresponding scores.

Table 5: Comparative labels and corresponding scores

Comparative score system	Comparative labels
2	Much higher
1	Higher
0	Same
-1	Lower
-2	Much lower

We put the comparative scores as follows: the score will be "1" or "-1" if the difference between the relative scores of two resumes is 1 or 2. The score will be "2" or "-2" if the difference between the relative scores of the two resumes is more than 2, and the score will be "0" if the relative scores of the two resumes are the same. As a result, resume A will be compared to each of the other resumes, one by one, per attribute.

Note that when we compare resumes A and B, we do not need the opposite comparative label comparing B and A because we only compare the possible pairwise combinations, not the permutations. The combination relations refer to the combination of elements  $n$  taken  $k$ -combination at a time without repetition, and the order of elements selection does not matter here. Therefore, a  $k$ -combination of the set  $S$  is hence a subset of  $k$  different elements from  $S$ . If there are  $n$  elements in the set, the number of  $k$ -combinations is equal to:

$$\frac{n!}{k!(n-k)!} \quad (2)$$

whenever  $k \leq n$ , and which is zero when  $k > n$

## 7.4 Ranking SVM Phase

Using the ranking SVM algorithm, the final resume ranking per semantic attribute will be based on comparative labels concerning the other N-1 resumes. Rank SVM is a technique for sorting lists of objects using pairwise difference vectors to adaptively arrange given comparable peers based on a certain criterion. Rank SVM performs rankings using Standard SVM. The ranking SVM's objective is to sort the list of resumes for each attribute, and we can infer the ordering list of resumes for all attributes. This ranking will enable further search capabilities based on how relevant the retrieved objects are to a particular search query.

In our work, we used (**train\_test\_split**) function of sklearn that split input resumes and values data into train and test dataset according to **test\_size** ratio. The **test\_size** is equal to 0.33 mean, then 33% is test data, and 67% is train data. The output is **resume\_train**, **resume\_test**, which is (**x\_train x\_test**) and (**y\_train, y\_test**). All resume comparisons are placed as pairwise inputs into the SVM ranking algorithm for the training, with each attribute labeled. We used hyper-parameter optimization in SVM for both C and Gamma values to get their optimized values. Then, pass training data to SVM to predict the comparative labels against each attribute and resume pairs by using the three following functions [58, 59]:

- **svm = SVC(kernel = 'rbf', c=10, gamma=0.1)**
- **svm.fit(x\_train, y\_train)**
- **predict\_eachAttribute = svm.predict(x\_test).**

Rank resumes based on predicted ranks so that in each iteration, the resume with the highest rank is selected and removed for the next iteration. Ultimately, we will get the ranks of resumes based on scores against individual attributes. Finally, the output will be the corresponding usable relative measurements and the ordering list based on comparisons of all resumes.

This method helps HR query the list of candidates ranked based on single or multiple semantic attributes in their resumes. As such, we can apply a search to retrieve the resume that is "Much higher" than the others in a certain attribute. To rearrange the resumes based on the resulting rankings, we can count (sum) their nascent relative measurement scores based on multiple or all attributes for each resume. The ranking is given in descending order to present the resumes from the best to the worst. A soft-margin ranking SVM method is used for a given set of attributes  $A$  to learn a ranking linear function  $r_a$  for each attribute, similar to the way used in [4]:

$$r_a(x_i) = w_i^T x_i \quad (3)$$

Where  $w_a$  is the coefficient of the ranking function  $r_a$  and  $x_i$  is a feature vector of attributes of a resume being ranked. Rearranging a set of comparisons into two groups can be considered a representation of the pairwise relative constraints needed to learn a ranking function. The first group is a set of dissimilarity comparisons  $D_a$  of ordered pairs so that  $(i, j) \in D_a \Rightarrow i > j$ . The second group is a set of similarity comparisons  $S_a$  of non-ordered pairs so that  $(i, j) \in S_a \Rightarrow i = j$ . Then, the following formula is used to get the  $w_a$

coefficients of  $r_a$  from the  $D_a$  and  $S_a$  sets:

$$\begin{aligned} & \text{minimize} \quad \left( \frac{1}{2} \|w_a^t\|^2 + C \sum \xi_{ij}^2 \right) \\ & \text{subject to} \quad w_a^T(x_i - x_j) \geq 1 - \xi_{ij}; \quad \forall (i, j) \in D_a \\ & \quad \quad \quad |w_a^T(x_i - x_j)| \leq \xi_{ij}; \quad \forall (i, j) \in S_a \\ & \quad \quad \quad \xi_{ij} \geq 0 \end{aligned} \quad (4)$$

$\xi_{ij}$  is the misclassification bias, and  $C$  is the trade-off between maximization of margin and minimization error. The resulting optimal  $w_a$  function can then be used to (explicitly) rank all training samples according to  $a$ . "Equation(3) is used to map a feature vector  $x_i$  to a feature vector consisting of several real-value relative measurements."

## 7.5 Performance evaluation

For SVM performance evaluation, we assess the accuracy of the proposed model, along with a classification report that quantifies the quality of the predictions of each semantic attribute.

A classification report assesses the quality of a classification algorithm's predictions. How many predictions are correct, and how many are incorrect? True Positives (TP), False Positives (FP), True Negatives (TN), and False Negatives (FN), to be exacted, as in Figure 3 [60].

Actual	Positive	TP	FN
	Negative	FP	TN
		Positive	Negative
		Predicted	

Figure 3: Simple representation of confusion matrix [60]

The input of the classification report is (**X, Y**) where X is predicted values and Y is actual values. We used (**classification\_report**) function of sklearn library to print precision, recall, and F1-score. Similarly, used (**accuracy\_score**) function of sklearn library to print the accuracy of the SVM model. Whereas the accuracy is the proportion of accurate predictions to the total number of data samples [61]. The precision shows how many predictions for a certain class are of the same class. The recall shows how many of certain class predictions were right, and the F1-score is the geometric mean of precision and recall [62]. The numeric implementation of the classification report is proven in the next section of the experiment results.

## 7.6 Practical Example of the Three Proposed Phases

The graph in Figure 4 shows our approach example with the following information:

- Inputs: 3 resumes (resume A, resume B, and resume C) with semantic education attributes extracted from resumes.
- Processing: Describing the relative and comparative labels with

corresponding scores for comparing between resumes.

-Output: Extract usable relative measurements in which these values are set and adjusted based on comparing each resume with the other resumes. Then, get the ordering list of all resumes using the SVM ranking algorithm.

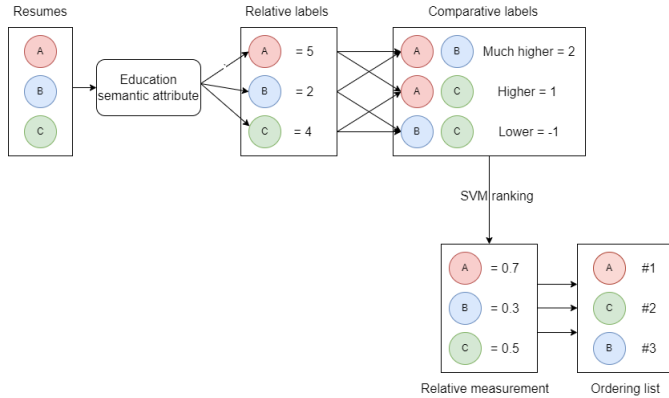


Figure 4: A simple example clarifies the main three phases of our proposed approach

## 8 Experiments and Results

### 8.1 Dataset

Our proposed approach’s experiment was conducted on a dataset containing 228 resumes in various text file formats, such as doc, docx, and pdf. We collected resumes from various technical (computer-related) specialists from various sources, including Github, Kaggle, LinkedIn, etc. Most resumes contain information about technical skills, such as programming in Python or Java, and other technical experience. We used the resume dataset for analysis, and each resume was assigned relative and comparative labels.

### 8.2 Implementation of three proposed phases

After the three phases of our approach, we generated 25,878 resume comparisons between resumes for each attribute. To select the candidate who matches the job requirements, we aggregated the scores and ranked the resumes according to each attribute and all attributes, as detailed in the methodology chapter. The resumes were ranked in descending order of total score, with the highest resume coming first and the lowest coming last. Next, we analyzed the results using four comparison-related aspects:

1. Ranking based on the regular scores. Figure 5 shows the histogram distribution of resume rankings for all attributes based on the regular scores.
2. Ranking based on the comparative scores. Figure 6 shows the histogram distribution of resume rankings based on the comparative scores for all attributes.
3. Ranking based on the comparative scores and regular scores. Figure 7 shows the histogram distribution of resume rankings for all attributes based on the comparative and regular scores.

4. Ranking based on the comparative scores and relative scores. Figure 8 shows the histogram distribution of resume rankings for all attributes based on the comparative and relative scores.

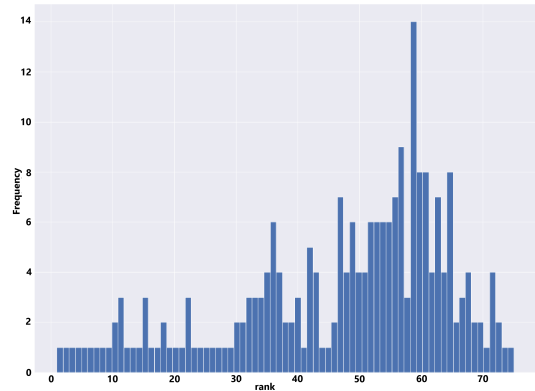


Figure 5: Resume rankings for all attributes based on the regular scores

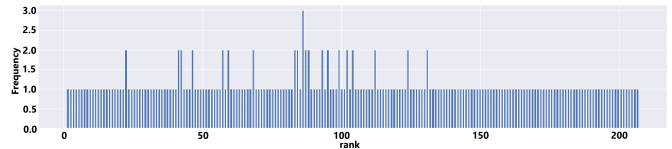


Figure 6: Resume rankings for all attributes based on the comparative scores

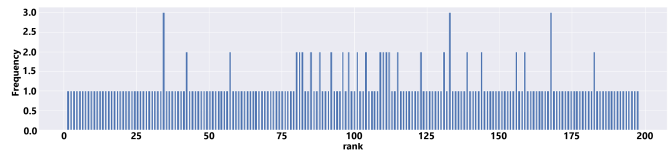


Figure 7: Resume rankings for all attributes based on the comparative scores and regular scores

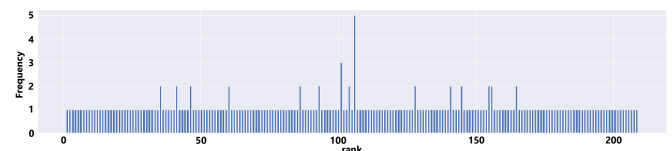


Figure 8: Resume rankings for all attributes based on the comparative scores and relative scores

### 8.3 Analysis of Results

Table 6 summarizes the four ranking results. The table compares the effectiveness of distributing and discriminating in the rankings of resumes with high unique values. It is observed that all three rankings based on comparative scores outperform the ranking based on regular scores, which offers very weak resume differentiation with high (undesired) redundant values and highly (confusable) similarities in resumes’ scores. It is important to observe that when it comes to resume distributions (as indicated in Table 6, Figure

6, Figure 8, and Figure 7), the more unique scores, the better the ranking result.

Table 6: Results of rankings with different four basis

Basis of rankings	Max Rank	No. of resumes that has unique scores	No. of resumes that has redundant scores
Regular scores	75	30	198
Comparative scores	207	187	41
Comparative and Regular	198	171	57
Comparative and Relative	<b>209</b>	<b>194</b>	<b>34</b>

Let us demonstrate the theoretical distribution of the duplicated rankings between resumes on each basis of rankings. Figure 9 shows the histogram distribution of the duplicated rankings between resumes for all attributes based on the regular scores. Figure 10 shows the histogram distribution of the duplicated rankings between resumes for all attributes based on the comparative scores. Figure 11 shows the histogram distribution of the duplicated rankings between resumes for all attributes based on the comparative and regular scores. Figure 12 shows the histogram distribution of the duplicated rankings between resumes for all attributes based on the comparative and relative scores.

As we proved in Table 6, the distribution of the duplicated rankings between resumes for all attributes based on the comparative and relative scores outperforms all other ranking bases.

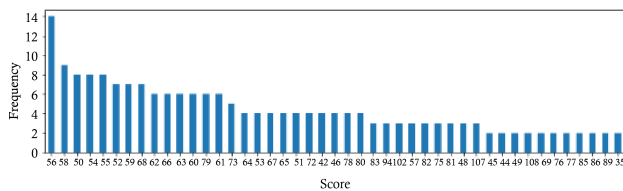


Figure 9: Duplicated rankings between resumes for all attributes based on the regular scores.

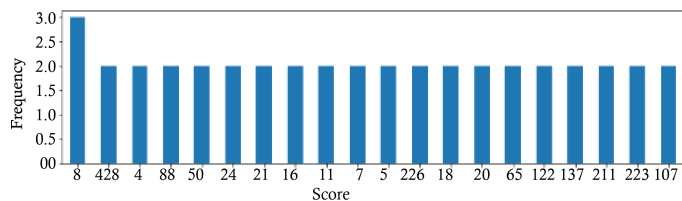


Figure 10: Duplicated rankings between resumes for all attributes based on the comparative scores.

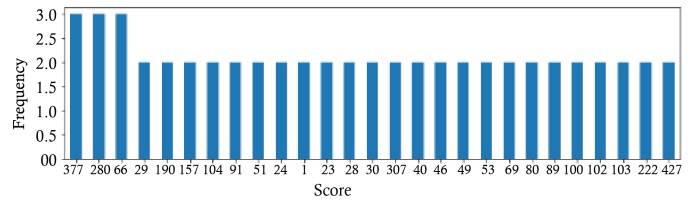


Figure 11: Duplicated rankings between resumes for all attributes based on the comparative and regular scores.

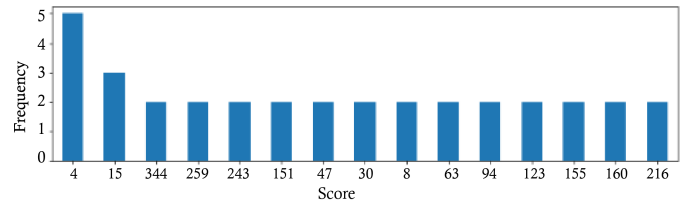


Figure 12: Duplicated rankings between resumes for all attributes based on the comparative and relative scores.

We proved that the method of ranking based on comparative and relative scores is very effective by comparing its accuracy with all other basis rankings as shown in Table 7. The ranking quality can further be measured as the accuracy of differentiating (as much as possible) between  $N$  of compared resumes. Resulting in as many as possible different unique scores  $S$ , which is, in the best-case scenario, equal to the number of compared resumes (i.e.,  $S = N$ ), implies that each resume has a unique (distinct) score. This approach is made by computing the percentage of the total sum of all unique and redundant scores concerning their frequencies using the following formula:

$$Accuracy = \frac{1}{N} \left( \sum_i^S \frac{1}{f_i} \right) * 100 \tag{5}$$

Where  $N$  is the number of resumes to be ranked.  $S$  is the total number of unique scores assigned to  $N$  number of resumes to be ranked, and  $f_i$  is the corresponding  $i$ th frequency of the  $i$ th unique score.

Table 7: Accuracy of rankings with different four basis

Basis of rankings	Accuracy
Regular scores	33%
Comparative scores	90%
Comparative and Regular scores	87%
Comparative and Relative scores	<b>92%</b>

As such, it is seen that all methods that rank resumes using comparative scores outperform those that rely on regular scores for ranking. The highest accuracy in ranking the resumes is achieved using ranks derived from comparative scores with the relative scores as the basis. This approach can be applied to each attribute. Figure 13 shows the rankings' histogram of the resumes for the personal information attribute, taking the comparative and relative scores as the basis.

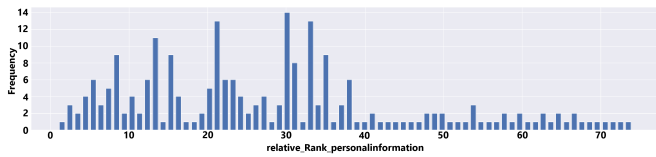


Figure 13: Resume rankings for personal information attribute based on the comparative scores and relative scores

Finally, Figure 14 depicts the top ten resumes’ (candidates) rankings with totaling scores on a comparative and relative basis.

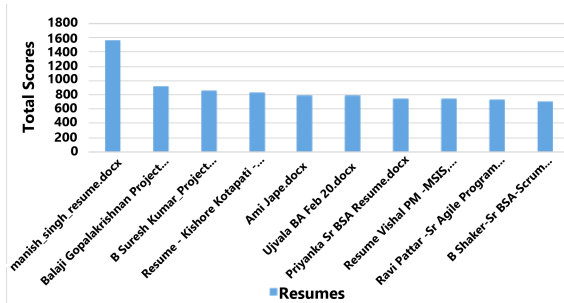


Figure 14: Top ten resume’ rankings

### 8.4 Evaluation of Resume Ranking per Attribute

Table 8 reports the accuracy [61] and classification report that contains a precision [63], recall [64], and F1 score [65] of the SVM ranking algorithm which quantifies the quality of the predictions of each of all semantic attributes.

Table 8: Evaluation metrics of the SVM ranking algorithm for each of all semantic attributes

Semantic Attribute	Accuracy	Precision	Recall	F1-Score
1- Personal information	99.168%	0.97751	0.97162	0.97456
2- Education level	99.496%	0.98852	0.99625	0.99238
3- Technical skills	99.461%	0.99106	0.98162	0.98634
4- Professional experience	99.754%	0.99516	0.99756	0.99636
5- Personal skills	99.841%	0.98810	0.99747	0.99278
6- Awards	99.660%	0.98212	0.99108	0.98660
7- Hobbies and interest	99.332%	0.98624	0.96987	0.97805
8- Additional Qualification	99.601%	0.98980	0.96697	0.97838
9- Professional Certificate	99.344%	0.97351	0.99453	0.98402
10- Suitable age requirement	99.941%	0.99099	0.99979	0.99539
11- Career objective	99.976%	0.99984	0.99963	0.99973
12- Project Experience	99.742%	0.98223	0.99774	0.98998
13- Languages	99.964%	0.99964	0.99975	0.99969
<b>Average score</b>	<b>99.652%</b>	<b>0.98805</b>	<b>0.98952</b>	<b>0.98878</b>

Through the SVM algorithm performance table, we note that the career objective attribute is higher than other attributes in accuracy, precision, and F1-score, and the suitable age requirement attribute is higher than other attributes in the recall. In general, the average values of all evaluation metrics represent high performance up to 99%. The table results prove the effectiveness of using an SVM algorithm to rank resumes for each semantic attribute separately.

## 9 Conclusions

This study presents a model to improve the candidate ranking and selection process for career matching. The model consists of three phases: firstly, the extraction of semantic information for 13 proposed attributes using NLP tools; secondly, inferred of relative and comparative descriptions by comparing every two resumes for each attribute; and finally, the use of the ranking SVM algorithm to order all resumes for each attribute. Implementing a comparison description in the candidates’ selection process has been suggested and proven more accurate and informative. It helps to distinguish the tiny differences between resumes. The experiment results were analyzed using four comparison aspects: ranking based on regular scores, comparative scores, comparative and regular scores, and comparative and relative scores. The histogram results prove that all three ranking methods based on comparative scores with high unique rankings values outperform the ranking using mere regular scores, which offers very weak resume differentiation with high redundant rankings values and high similarities in resume scores. Also, the ranking attained by the comparative scores with the relative scores as the basis achieves the highest accuracy of 92%. We reported evaluation metrics of the SVM ranking algorithm with an average accuracy of 99.652%, an average precision of 0.98805, an average recall of 0.98952, and an average F1-score of 0.98878, which quantifies the quality of the predictions of each of all semantic attributes. Finally, we have obtained a list of the top ten candidates out of 228 technical specialists’ resumes.

## 10 Future Work

In the future, a proposed model should consider keyword matching for 13 attributes and focus on the semantic relations and meanings of the resume language to account for variations in resume authoring. We will explore additional parts that individuals may incorporate into their resumes to enhance the evaluation and comparison process for more effective candidate-career matching. Furthermore, we will explore the possibility of applying the proposed methods to many domains beyond the technological field and incorporating languages other than English, such as Arabic. Finally, we will evolve our work to analyze the job description document to calculate its matching with the resume as another way different from calculating evaluation metrics of job requirements.

**Acknowledgment** This work is selected as a special issue of a conference paper published in the 2022 14th International Conference on Computational Intelligence and Communication Networks (CICN) to be extended into a journal paper.

## References

- [1] A. H. Alderham, E. S. Jaha, “Comparative Semantic Resume Analysis for Improving Candidate-Career Matching,” in 2022 14th International Conference on Computational Intelligence and Communication Networks (CICN), 313–321, IEEE, 2022, doi:10.1109/CICN56167.2022.10008255.
- [2] Y. Lin, H. Lei, P. C. Addo, X. Li, “Machine learned resume-job matching solution,” arXiv preprint arXiv:1607.07657, 2016.



- [3] C. Saxena, "Enhancing productivity of recruitment process using data mining & text mining tools," 2011.
- [4] E. S. Jaha, M. S. Nixon, "Soft biometrics for subject identification using clothing attributes," in *IEEE international joint conference on biometrics*, 1–6, IEEE, 2014, doi:10.1109/BTAS.2014.6996278.
- [5] D. Lee, M. Kim, I. Na, "Artificial intelligence based career matching," *Journal of Intelligent & Fuzzy Systems*, **35**(6), 6061–6070, 2018, doi:10.3233/JIFS-169846.
- [6] L. Columbus, "Remote Recruiting In A Post COVID-19 World," <https://www.forbes.com/sites/louiscolumbus/2020/03/30/remote-recruiting-in-a-post-covid-19-world/?sh=7c4ce7e599fd>, 2020.
- [7] C. Russell, *Right Person–Right Job: Guess Or Know: the Breakthrough Technologies of Performance Information, Human Resource Development*, 2002.
- [8] A. Swenson, "The new art of hiring smart: Matching the right person to the right job," Technical report, 2000.
- [9] A. K. Sinha, M. A. K. Akhtar, A. Kumar, "Resume Screening Using Natural Language Processing and Machine Learning: A Systematic Review," *Machine Learning and Information Processing: Proceedings of ICMLIP 2020*, **1311**, 207, 2021.
- [10] Y. Suresh, A. M. Reddy, et al., "A Contextual Model for Information Extraction in Resume Analytics Using NLP's Spacy," in *Inventive Computation and Information Technologies*, 395–404, Springer, 2021, doi:10.1007/978-981-33-4305-4\_30.
- [11] B. Gaur, G. S. Saluja, H. B. Sivakumar, S. Singh, "Semi-supervised deep learning based named entity recognition model to parse education section of resumes," *Neural Computing and Applications*, **33**(11), 5705–5718, 2021, doi:10.1007/s00521-020-05351-2.
- [12] A. Kulkarni, A. Shivananda, "Advanced Natural Language Processing," in *Natural Language Processing Recipes*, 97–128, Springer, 2019.
- [13] H. Assal, J. Seng, F. Kurfess, E. Schwarz, K. Pohl, "Semantically-enhanced information extraction," in *2011 Aerospace Conference*, 1–14, IEEE, 2011.
- [14] S. Singh, "Natural language processing for information extraction," arXiv preprint arXiv:1807.02383, 2018.
- [15] D. Martinho-Corbishley, M. S. Nixon, J. N. Carter, "Analysing comparative soft biometrics from crowdsourced annotations," *IET Biometrics*, **5**(4), 276–283, 2016, doi:10.1049/iet-bmt.2015.0118.
- [16] D. A. Reid, M. S. Nixon, S. V. Stevenage, "Soft biometrics; human identification using comparative descriptions," *IEEE Transactions on pattern analysis and machine intelligence*, **36**(6), 1216–1228, 2013, doi:10.1109/TPAMI.2013.219.
- [17] S. Bird, E. Klein, E. Loper, *Natural language processing with Python: analyzing text with the natural language toolkit*, "O'Reilly Media, Inc.", 2009.
- [18] D. Parikh, K. Grauman, "Relative attributes," in *2011 International Conference on Computer Vision*, 503–510, IEEE, 2011.
- [19] N. Y. Almodhahka, M. S. Nixon, J. S. Hare, "Comparative face soft biometrics for human identification," in *Surveillance in Action*, 25–50, Springer, 2018, doi:10.1007/978-3-319-68533-5\_2.
- [20] J. Chen, C. Zhang, Z. Niu, "A two-step resume information extraction algorithm," *Mathematical Problems in Engineering*, **2018**, 2018, doi:10.1155/2018/5761287.
- [21] L. Pham Van, S. Vu Ngoc, V. Nguyen Van, "Study of Information Extraction in Resume," Conference, 2018.
- [22] R. Nimbekar, Y. Patil, R. Prabhu, S. Mulla, "Automated Resume Evaluation System using NLP," in *2019 International Conference on Advances in Computing, Communication and Control (ICAC3)*, 1–4, IEEE, 2019, doi:10.1109/ICAC347590.2019.9036842.
- [23] Y. O. German, O. V. German, S. Nasr, "Information extraction method from a resume (CV)," 2019.
- [24] S. Amin, N. Jayakar, M. Kiruthika, A. Gurjar, "Best Fit Resume Predictor," *International Research Journal of Engineering and Technology*, **6**(8), 813–820, 2019.
- [25] S. T. Gopalakrishna, V. Vijayaraghavan, "Automated Tool for Resume Classification Using Sematic Analysis," *International Journal of Artificial Intelligence and Applications (IJAIA)*, **10**(1), 2019.
- [26] V. Bhatia, P. Rawat, A. Kumar, R. R. Shah, "End-to-end resume parsing and finding candidates for a job description using bert," arXiv preprint arXiv:1910.03089, 2019.
- [27] Y.-C. Chou, H.-Y. Yu, "Based on the application of AI technology in resume analysis and job recommendation," in *2020 IEEE International Conference on Computational Electromagnetics (ICCEM)*, 291–296, IEEE, 2020, doi:10.1109/ICCEM47450.2020.9219491.
- [28] P. Kadu, V. Z. Ashwini, "Knowledge Extraction from Text Document Using Open Information Extraction Technique," *International Journal of Advanced Trends in Computer Science and Engineering (IJATCSE) Volume*, **9**, 2020.
- [29] F. O. Oladipo, A. A. Ayomikun, "A MODEL FOR AUTOMATIC RESUME SUMMARIZATION," *Journal of Information Systems & Operations Management*, **14**(2), 147–159, 2020.
- [30] P. K. Roy, S. S. Chowdhary, R. Bhatia, "A Machine Learning approach for automation of Resume Recommendation system," *Procedia Computer Science*, **167**, 2318–2327, 2020.
- [31] K. Satheesh, A. Jahnavi, L. Iswarya, K. Ayesha, G. Bhanusekhar, K. Hanisha, "Resume Ranking based on Job Description using SpaCy NER model," *International Research Journal of Engineering and Technology (IRJET) e-ISSN:* 2395–0056, **7**(05), 2020.
- [32] V. Mittal, P. Mehta, D. Relan, G. Gabrani, "Methodology for resume parsing and job domain prediction," *Journal of Statistics and Management Systems*, **23**(7), 1265–1274, 2020, doi:10.1080/09720510.2020.1799583.
- [33] N. Khamker, "Resume Match System," *International Journal of Innovative Science and Research Technology*, 2021.
- [34] S. Bhor, H. Shinde, V. Gupta, V. Nair, M. Kulkarni, "Resume parser using natural language processing techniques," 2021.
- [35] R. Haddad, E. Mercier-Laurent, "Curriculum Vitae (CVs) Evaluation Using Machine Learning Approach," in *IFIP International Workshop on Artificial Intelligence for Knowledge Management*, 48–65, Springer, 2021.
- [36] D. Vukadin, A. S. Kurdija, G. Delač, M. Šilić, "Information Extraction from Free-Form CV Documents in Multiple Languages," *IEEE Access*, 2021.
- [37] K. F. F. Jiechieu, N. Tsope, "Skills prediction based on multi-label resume classification using CNN with model predictions explanation," *Neural Computing and Applications*, **33**(10), 5069–5087, 2021.
- [38] J. Zhao, J. Wang, M. Sigdel, B. Zhang, P. Hoang, M. Liu, M. Korayem, "Embedding-based Recommender System for Job to Candidate Matching on Scale," arXiv preprint arXiv:2107.00221, 2021.
- [39] K. Tejaswini, V. Umadevi, S. M. Kadiwal, S. Revanna, "Design and Development of Machine Learning based Resume Ranking System," *Global Transitions Proceedings*, 2021.
- [40] K. Bothmer, T. Schlippe, "Skill scanner: connecting and supporting employers, job seekers and educational institutions with an AI-based recommendation system," in *The Learning Ideas Conference*, 2022.
- [41] I. Ali, N. Mughal, Z. H. Khand, J. Ahmed, G. Mujtaba, "Resume classification system using natural language processing and machine learning techniques," *Mehran University Research Journal Of Engineering & Technology*, **41**(1), 65–79, 2022.
- [42] B. Amro, A. Najjar, M. Macido, "An Intelligent Decision Support System For Recruitment: Resumes Screening and Applicants Ranking," 2022.

- [43] B. Kinge, S. Mandhare, P. Chavan, S. Chaware, "Resume Screening Using Machine Learning and NLP: A Proposed System," 2022.
- [44] O. Dada, A. Kana, S. Abdullahi, "An ontology based approach for improving job search in online job portals," *Journal of Computer Science and Its Application*, **25**(1), 34–43, 2018.
- [45] M.-I. Enăchescu, "Screening the Candidates in IT Field Based on Semantic Web Technologies: Automatic Extraction of Technical Competencies from Unstructured Resumes." *Informatica Economica*, **23**(4), 2019.
- [46] B. Kelkar, R. Shedbale, D. Khade, P. Pol, A. Damame, "Resume Analyzer Using Text Processing," *Journal of Engineering Sciences*, 353–361, 2020.
- [47] A. Wosiak, "Automated extraction of information from Polish resume documents in the IT recruitment process," *Procedia Computer Science*, **192**, 2432–2439, 2021.
- [48] T. T. Phan, V. Q. Pham, H. D. Nguyen, A. T. Huynh, D. A. Tran, V. T. Pham, "Ontology-based resume searching system for job applicants in information technology," in *International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems*, 261–273, Springer, 2021.
- [49] H. Sajjid, J. Kanwal, S. U. R. Bhatti, S. A. Qureshi, A. Basharat, S. Hus-sain, K. U. Khan, "Resume Parsing Framework for E-recruitment," in *2022 16th International Conference on Ubiquitous Information Management and Communication (IMCOM)*, 1–8, IEEE, 2022.
- [50] N. Tülümen, G. S. A. Genç, A. Öztaş, T. Güngör, A. Nohutcu, "Ontology based job and resume matcher," .
- [51] M. N. Uddin, T. H. Duong, K.-J. Oh, J.-G. Jung, G.-S. Jo, "Experts search and rank with social network: An ontology-based approach," *International Journal of Software Engineering and Knowledge Engineering*, **23**(01), 31–50, 2013.
- [52] I. Abuhassan, A. M. AlMashaykhi, "Domain ontology for programming lan-guages," *Journal of Computations & Modelling*, **2**(4), 75–91, 2012.
- [53] L. Mostafa, S. Beshir, "Job candidate rank approach using machine learn-ing techniques," in *International Conference on Advanced Machine Learning Technologies and Applications*, 225–233, Springer, 2021.
- [54] D. Parikh, A. Kovashka, A. Parkash, K. Grauman, "Relative attributes for enhanced human-machine communication," in *Proceedings of the AAAI Con-ference on Artificial Intelligence*, volume 26, 2153–2159, 2012.
- [55] A. Sainani, P. Reddy, "Extracting special information to improve the efficiency of resume selection process," 2011.
- [56] F. Yasmin, M. I. Nur, M. S. Arefin, "Potential candidate selection using informa-tion extraction and skyline queries," in *International Conference on Computer Networks, Big data and IoT*, 511–522, Springer, 2019.
- [57] O. Pathak, "A simple resume parser used for extracting information from resumes," <https://omkarpathak.in/pyresparser/>, 2019.
- [58] "sklearn.svm.SVC," <https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html>.
- [59] "Scikit Learn - Support Vector Machines," [https://www.tutorialspoint.com/scikit\\_learn/scikit\\_learn\\_support\\_vector\\_machines.htm](https://www.tutorialspoint.com/scikit_learn/scikit_learn_support_vector_machines.htm).
- [60] MLee, "Visual Guide to the Confusion Ma-trix," <https://towardsdatascience.com/visual-guide-to-the-confusion-matrix-bb63730c8eba>, 2021.
- [61] D. Cournapeau, "sklearn.metrics.accuracy\_score," [https://scikit-learn.org/stable/modules/generated/sklearn.metrics.accuracy\\_{score}.html](https://scikit-learn.org/stable/modules/generated/sklearn.metrics.accuracy_{score}.html), 2007-2022.
- [62] S. Solanki, "Model Evaluation & Scoring Metrics," <https://coderzcolumn.com/tutorials/machine-learning/model-evaluation-scoring-metrics-sikit-learn-sklearn>.
- [63] D. Cournapeau, "sklearn.metrics.precision\_score," [https://scikit-learn/stable/modules/generated/sklearn.metrics.precision\\_{score}.html](https://scikit-learn/stable/modules/generated/sklearn.metrics.precision_{score}.html), 2007-2022.
- [64] D. Cournapeau, "sklearn.metrics.recall\_score," [https://scikit-learn/stable/modules/generated/sklearn.metrics.recall\\_{score}.html](https://scikit-learn/stable/modules/generated/sklearn.metrics.recall_{score}.html), 2007-2022.
- [65] D. Cournapeau, "sklearn.metrics.f1\_score," [https://scikit-learn/stable/modules/generated/sklearn.metrics.f1\\_{score}.html](https://scikit-learn/stable/modules/generated/sklearn.metrics.f1_{score}.html), 2007-2022.

**Copyright:** This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY-SA) license ( <https://creativecommons.org/licenses/by-sa/4.0/>).