Job Recommendation System Based on Different Skill Extraction Methods

Ziyun Yu and **Yiru Zhou** and **Jiarui Shi** and **Jiayi Xu**
New York University
zy2478@nyu.edu and yz8297@nyu.edu and js12566@nyu.edu and jx2325@nyu.edu

Abstract

Our project aims to create an efficient job recommendation system that leverages user profiles to provide personalized job recommendations. The primary motivation is to assist job seekers in finding opportunities that align with their skills and interests. The project is structured into two main components: term extraction of both job descriptions and resumes, and a recommendation system based on calculated similarities. In terms of the term extraction part, we apply different methods on a dataset of over 6000 job descriptions (eg. integration of CRF and BIO TAG (Green et al., 2022), SkillNer[1], etc) and on a dataset of over 29000 resumes (Jiechieu and Tsopze, 2021) (eg. the Pyresparser package [2], a pre-trained skill NER model[3], and SkillNer. We evaluate the performance of different methods we apply, where the test set is generated by a manual annotation process on a subset of 100 job descriptions and resumes respectively. We use soft f1-score, soft recall, and soft precision (Fränti and Mariescu-Istodor, 2023) specifically for the evaluation of the performance of different keyword extraction methods. Subsequently, we pair each attempted job description skill extraction method with each resume skill extraction method. We access the extraction performance for each pairing, compared the results, and build a recommendation system accordingly.

1 Introduction

Navigating today's ever-changing job landscape is more challenging than ever. With expanding and diverse job markets, a reliable guide is essential for individuals to discover roles that match their skills. In today's world, although we already have platforms like Linkedin, in this sea of opportunities, the process of finding the right job remains a difficult effort. Job seekers face the daunting challenge of sifting through a multitude of postings, while employers strive to identify candidates whose skills align with their businesses' culture.

Our motivation for embarking on this research stems from a genuine desire to simplify this intricate dance between job seekers and employers. Recognizing the value of a well-designed job recommendation system, we intend to improve the search experience by making it more personalized, efficient, and rewarding. By analyzing the latest NYC Open Data job listings[4], our goal is to understand the current job market dynamics and address the practical needs of both job seekers and employers.

This paper introduces a two-phase strategy to tackle job recommendation complexities. The first phase focuses on extracting key terms of skills from job descriptions and profiles, crucial for accurate and meaningful recommendations. In the second phase, we build a recommendation system that uses these terms to provide tailored job suggestions, aligning with the aspirations and capabilities of job seekers.

2 Related Works

Skills extraction, the process of identifying and extracting relevant skills from textual data, has garnered significant attention in recent years due to its applications in various domains such as recruitment, talent management, and educational assessment. In this section, we provide an overview of the related works in skill extraction, spanning from early methodologies to more recent advancements.

Zhang et al. (2022) introduced a new dataset, SkillSpan, specifically tailored for skill extraction, enriching it with classifications of hard and soft skills annotated by domain experts. In the appli-

[1] https://github.com/AnasAito/SkillNER/tree/master.

[2] https://github.com/OmkarPathak/pyresparser?tab=readme-ov-file

[3] https://github.com/Shavakchauhan/Resume_parser_using_deep_learning

[4] https://data.cityofnewyork.us/City-Government/Jobs-NYC-Postings/kpav-sd4t.

cation of SkillSpan, they observed that domain-adapted models outperformed their non-adapted counterparts. In our work, we draw inspiration from their categorization of soft and hard skills, particularly in the sections related to resume skill extraction and the evaluation of recommendation systems.

Recently, Named Entity Recognition (NER) (Mohit, 2014) has many applications in skill extraction. Regarding this method, a recent paper with high recognition is "*SkillNER: Mining and Mapping Soft Skills from any Text*". Fareri et al. (2021) manually annotate the corpus with entities in two steps, clue extraction and skill extraction. Clue is defined as a set of terms, lexical expressions, or recurrent patterns correlated with the appearance of the soft skill. For the clue extraction, they manually build a seed list of soft skills first and then compile a list of documents related to those skills. After that, they automatically collect the soft skills extraction contexts using a rule-based matcher in spaCy. Lastly, they extract the list of clues from the collected extraction contexts. For the part of skills extraction, they search the list of clues in their corpus of soft skills and then include a group of experts to manually annotate the sentences corresponding to the clues found. After they get the annotated soft skill corpus, they train two NER models using a feature-based support vector machine and a deep learning method respectively. In our work, NER is also a significant method that we implement several methods about NER. However, different from Silvia Fareri, our focus is more on non-machine learning methods, such as N-grams, when implementing NER. Specifically, we use a pre-trained model of the same name as Fareri's SkillNER but actually not the same approach.

In addition to the advancements in skills extraction methodologies discussed earlier, it is noteworthy to highlight the contribution of the study "*Skills prediction based on multi-label resume classification using CNN with model predictions explanation*" (Jiechieu and Tsopze, 2021), which demonstrated the efficacy of Convolutional Neural Networks (CNN) in predicting requisite skills for specific job titles using anonymous resumes. By leveraging this proven tool alongside a multi-label classification architecture based on CNN, they aim to not only extract explicitly mentioned skills but also predict high-level skills from resumes, even when they are not explicitly stated. This pivotal research not only reinforces the viability of CNN in skill prediction but also introduces a practical tool, the Pyresparser package, designed for extracting skills from unstructured textual data, particularly from accessible anonymous resumes. The incorporation of the Pyresparser package into our project aligns seamlessly with our overarching goal of enhancing skills extraction processes. It is definitely one of the potential methods we could try to extract skills from the resumes.

Green et al. (2022) tried using the combination of BIO and CRF to extract skills entities from the job description data while distinguishing between the concepts of skills, qualifications, experience, and domains. They trained their model using CRF based on the BIO-tagged corpus to extract the skills. This enlightened us to not only test their method, but also make improvements based on their method, such as in the way they did sentence segmentation. In their experiment, they used NLTK for sentence segmentation. The possible reason is that their dataset is in an organized manner and every word is spelled and separated correctly. However, NLTK is likely not to be the best choice when dealing with unstructured textual content. Trying to find a better way to conduct sentence segmentation is challenging but meaningful in our project.

3 Methodology

In this section, we describe how we select the materials for resumes and job postings, the different methods of terminology extraction, and how we get the relevance between resumes and job postings.

3.1 Data Selection

Our dataset is divided into two main parts: anonymous resumes and job postings. Below, we will provide detailed explanations regarding the sources of each segment.

3.1.1 Resume

The resume dataset used for our resume section is secured from the published paper written by Jiechieu and Tsopze (2021).

The dataset consists of 28,707 anonymous resumes sourced from indeed.com. These resumes, distributed across 10 IT-related classes, are extracted from various PDFs. It's worth noting that the resumes exhibit irregular structures, encompassing information about individuals' skills, past experiences, degrees, and more. The classes include competencies such as Software Developer, Front-End Developer, Network Administrator, Web Developer, Project Manager, Database Administrator, Security Analyst, Systems Administrator, Python Developer, and Java Developer. The data extraction process aims to capture diverse information despite the non-uniform structure of the resumes.

3.1.2 Job Posting

The dataset used for our job posting section is sourced from the NYC Open Data website[5]. This dataset provides comprehensive information about the latest job postings in New York, including details such as Job ID, Agency, Posting Type, Number of Positions, Business Title, Civil Service Title, Title Code Number, Level, Salary Range, Salary Frequency, Work Location, Division/Work Unit, Job Description, Minimum Qualification Requirements, Preferred Skills, Additional Information, Application Instructions, Hours/Shift, Work Location Details, Recruitment Contact, Residency Requirement, and Post Until. Specifically, we focus on the Minimum Qualification and Preferred Skills sections, comprising over 6,800 positions spanning various industries and enterprises. The content includes complete sentences; however, it is important to note the presence of non-alphabetic symbols and unclear sentence segmentation, necessitating substantial post-processing efforts.

3.1.3 Dataset Splitting

During the process of skills extraction validation, we randomly select 100 samples each from the resumes and job postings. In the meantime, when doing validation for the recommendation system, we select 100 resumes as the input but tested the output based on the overall job posting dataset. It ensures that the recommendation system has enough background job postings to choose from. Regarding the test set, we also randomly selected 100 resumes and all the job postings. The rest of the data are included in the training set.

3.2 Skills Extraction

In this section, we describe the implementation and effects of the specific terminology extraction. Models mentioned below are described in detail in the experiments section.

In defining skill-related vocabulary for our project, we have established the following criteria:

1. Phrases are permitted, but only those constituting strong, tightly bound expressions (such as "Microsoft Office," "cybersecurity," or "creative thinking") are recognized as skill-related phrases. Other instances are preferably split as individual words.

2. Determiners, such as "a" or "the", are ignored in the identification process.

3. Adjectives and adverbs denoting degrees of modification (e.g., “strong” in "strong organizational skills", “MS Office” in “basic MS Office skills”) are omitted from consideration, aiming to recognize the core skill (e.g., "MS Office”).

4. Terms related to educational qualifications (e.g., "master’s degree," "high school diploma") are included within the scope of skills.

5. Phrases indicating the ability to, knowledge of, or understanding of, such as "ability to," "knowledge of," are regarded as positional cues for skills but are not directly incorporated into the skill-related vocabulary.

3.2.1 Job Posting

The initial steps involve preprocessing the corpus into individual words with appropriate labels. Subsequently, two distinct models are applied to extract vocabulary related to skills from the processed data.

BIO + CRF In the preprocessing phase, we employ the robust natural language processing package, NLP-Cube. This versatile toolkit encompasses various functionalities, with a key focus on lexical paraphrasing and sentence segmentation. Opting for NLP-Cube over alternatives like NLTK is driven by the irregularities present in the dataset, where words exhibit non-standard forms. The paraphrasing capability of NLP-Cube contributes to the normalization of vocabulary, and its performance in sentence segmentation surpasses that of NLTK. Therefore, NLP-Cube emerges as a more suitable choice, providing enhanced efficiency and accuracy in data preprocessing. Its robust features significantly elevate the quality of the preprocessing pipeline in our academic investigation.

[5] https://data.cityofnewyork.us/City-Government/Jobs-NYC-Postings/kpav-sd4t.

For the next extraction step, inspired by the paper written by Green et al. (2022), we employ an advanced entity recognition methodology. Specifically, we utilize Conditional Random Fields (CRF), a sequence labeling algorithm, to perform structured prediction tasks. This approach aims to enhance the precision and granularity of information extraction.

To label the dataset, we employ the dataset from the paper which provides job descriptions with 4,917,794 items (sentences) in total. This dataset is highly credible because "Workers were required to pass a ’qualification task’ before they were assigned a bespoke qualification allowing them to contribute to the live task" (Green et al., 2022). Every word is labeled with B (before the targeted terms), I (inside the targeted terms), and O (outside the targeted terms) indicating entities such as skills, qualifications, occupations, and domains. For our project, because we do not set clear boundaries between skills, qualifications, occupations, and domains, we treat them all as skills. All of these terms can provide useful information regarding the matching between the resumes and the jobs. This labeling schema contributes to a more nuanced understanding of job qualification requirements.

The CRF model, trained over multiple epochs with both L1 and L2 regularization techniques, showcase a robust capability to extract detailed information from job qualification descriptions. The incorporation of a labeled dataset with BIO tags significantly improve the accuracy and specificity of the extracted entities.

Ultimately, using this model, we can successfully gain the skills vectors extracted from the job postings corpus.

SkillNer_Job We utilize the SkillNer package, an open-source Natural Language Processing (NLP) module, to automatically extract skills and certifications from unstructured job postings. One distinctive feature of SkillNer is its integration with the EMSI[6] database. This open-source skill database serves as a knowledge base linker, contributing to the prevention of skill duplications during the extraction process. By referencing the EMSI database, SkillNer ensures that identified skills are aligned with a standardized set, enhancing accuracy and minimizing redundancy in the extracted information.

[6] https://lightcast.io/open-skills

Original text	(1) Four (4) years of full-time, satisfactory experience in mechanical engineering work; and (2) A valid New York State Professional Engineerâ□□s License. Current New York State registration as a Professional Engineer must be maintained for the duration of your employment. A masters degree in mechanical engineering from an accredited college or university, accredited by regional, national, professional or specialized agencies recognized as accrediting bodies by the U.S. Secretary of Education and by the Council for Higher Education Accreditation (CHEA) may be substituted for one year of the mechanical engineering experience required in â□□1 above. Special Note: In addition to above qualification requirements, to be eligible for placement in Assignment Levels II and III, individuals must have at least one year within the last three years of experience as a major contributor or a project leader on a complex project requiring additional and specific expertise in the disciplines needed to design or construct the project. *Baccalaureate or Masterâ□□s degree in Mechanical Engineering. *Knowledge in NYC Codes and Standards, including NYC Building and Fire Code, NFPA 11, 12,13 14, 750, 2010. *At least 2 years of complex plan examination review. *Excellent written and communication skills.
BIO+CRF extraction	['satisfactory experience in mechanical engineering work', "2) a valid New York State Professional Enfiner 's license", 'professional engineer', 'master degree in mechanical engineering', 'professional', 'mechanical engineering', 'at little one year within the last three year of experience', 'design or construct the project .Baccalaureate', "master 's degree in mechanical engineering", 'knowledge in NYC code and standard', 'NYC Building and Fire code', 'complex plan examination review', 'write and communication skill']
SkillNer extraction	['mechanical engineering', 'professional engineer', 'mechanical engineering', 'high education', 'mechanical engineering', 'mechanical engineering', 'NFPA', 'professional', 'professional', 'license', 'registration', 'professional', 'addition', 'levels', 'additional', 'construct', 'code nfpa', 'plan examination', 'communication skills']

Figure 1: Job Postings Comparison: jobID = 533493

The SkillNer package is primarily built upon the utilization of a comprehensive English language model provided by spaCy, and a processed skills dictionary derived from the EMSI database. The workflow involves leveraging spaCy’s extensive language understanding capabilities to process textual input. This framework employs various matching techniques, including exact matching, abbreviation matching, and matching in low-confidence formats. To enhance the precision of skill predictions, the system incorporates a scoring mechanism based on n-gram analysis. This multi-faceted approach ensures a robust and versatile skill extraction process, contributing to the effectiveness of the SkillNer package in capturing a diverse range of skills from the input data.

Comparison In examining the current chart (figure 1), we observe the extraction results for the example with jobID = 533493 using both models. The disparity in the skill-related vocabulary extracted by these two models is noteworthy. The BIO + CRF model demonstrates a preference for capturing longer phrases with more complete sentence structures, often containing prepositions, as skills. Conversely, the SkillNer model tends to capture shorter compound phrases, typically consisting of 1-2 words, as skills.

The reason that the BIO + CRF model is more likely to extract longer phrases is that BIO functions better in determining the boundaries between different phrases and capturing the continuity inside the phrases. This ensures that phrases appear to be especially complete with all related information included. Furthermore, because CRF has strength in considering the contextual dependencies in a sequence and the global sequence context features, the phrases are prone to be extracted together with additional related information in the following context. Therefore, the BIO + CRF is featured capturing longer phrases with abundant related information. Accordingly, however, this strength is also possible to make the model capture too long phrases with comparatively useless information. This underscores the need for careful consideration and potential fine-tuning to strike a balance between capturing relevant details and avoiding the inclusion of superfluous information.

In comparing the output results of the SkillNer and BIO + CRF models, the distinct characteristics of SkillNer's output can be attributed to its heavy reliance on the skills processed from the EMSI database. Our dataset, however, may encompass skills that are absent from this database. This discrepancy arises for several reasons.

Firstly, our definition of skill-related terms includes some experience-related vocabulary, such as educational qualifications and certifications. Additionally, our definition explicitly excludes many adjectives denoting degrees of modification, like the term "professional" identified in this context.

Another contributing factor is that SkillNer solely relies on matching with the skills processed from the database. Despite the incorporation of n-grams for enhanced matching, the model still lacks contextual awareness, leading to instances where it fails to consider the broader context, resulting in the identification of "communication skill" while overlooking the complete phrase "write and communication skill." This underscores the model's dependence on pre-processed data and its potential limitations in recognizing skills beyond the confines of the EMSI database.

3.2.2 Resume

Three distinct approaches are employed to extract skills-related vocabulary from resumes.

Pyresparser We employ the Pyresparser package in Python for extracting skill-related terms from resumes. After processing with Pyresparser, the terminologies extracted from resumes include categories such as email, degree, skill, experience, company names, college, etc. In our analysis, we focus on the content extracted under the categories of degree, skill, and experience, which collectively constitute the results of our resume terminology extraction.

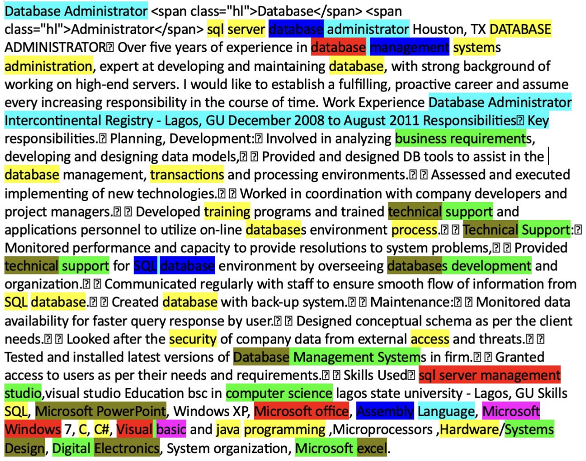
Pre-trained NER Another model that we use is a pre-trained model which was built for Named Entity Recognition on resumes. The model is trained on 551 resumes with existing entities including "JOBTITLE", "SKILL", "EXPERIENCE", "ORG", "TOOL", "DEGREE", and "EDUC". The model used a Robustly Optimized BERT Pretraining Approach (Roberta)[7], which is a transformer and was trained using the spaCy library with a configuration file, and the configuration file was autofilled with default values and subsequently saved as config.cfg. When the model runs on our dataset, the output consists of two parts, the extracted phrase of skills and the corresponding named entity. For the skills extraction part, we only take the first elements of the outputs and ignore the second elements, which are the entities.

SkillNer_Resume We also apply the SkillNer package to the dataset of resumes to extract terminologies. The details of the package have been mentioned above .

Due to the similarity in terms of contents between the resume corpus and the job descriptions, the application of the job description skills extraction method to the skills extraction of a resume is also desirable theoretically, and vice versa. So we make an attempt to apply the SkillNer model to job posting skill extraction to the resume skill extraction once again.

Comparison According to figure 2, the Pyresparser model generates numerous outputs, with the majority being single words. It does not handle duplicate words, but retains the plural form 's' in the original words, and uniformly converts all capitalizations to lowercase. The pre-trained NER model yields fewer but longer extraction results, predominantly centered around the domain of data management. These results exhibit a relatively singular focus. In contrast, the SkillNer model produces a greater number of shorter extraction results, covering a broader range of skills, including database, Microsoft, and system-related skills. However, its depth in specific domains, such as database expertise, is not as pronounced as the pre-trained NER model, despite its extensive breadth. For instance, the SkillNer model captures a comprehensive range

[7] https://huggingface.co/docs/transformers/model_doc/roberta



Database Administrator Database Administrator sql server database administrator Houston, TX DATABASE ADMINISTRATOR⍰ Over five years of experience in database management systems administration, expert at developing and maintaining database, with strong background of working on high-end servers. I would like to establish a fulfilling, proactive career and assume every increasing responsibility in the course of time. Work Experience Database Administrator Intercontinental Registry - Lagos, GU December 2008 to August 2011 Responsibilities⍰ Key responsibilities.⍰ Planning, Development:⍰ Involved in analyzing business requirements, developing and designing data models,⍰ ⍰ Provided and designed DB tools to assist in the database management, transactions and processing environments.⍰ ⍰ Assessed and executed implementing of new technologies.⍰ ⍰ Worked in coordination with company developers and project managers.⍰ ⍰ Developed training programs and trained technical support and applications personnel to utilize on-line databases environment process.⍰ ⍰ Technical Support:⍰ Monitored performance and capacity to provide resolutions to system problems,⍰ ⍰ Provided technical support for SQL database environment by overseeing databases development and organization.⍰ ⍰ Communicated regularly with staff to ensure smooth flow of information from SQL database.⍰ ⍰ Created database with back-up system.⍰ ⍰ Maintenance:⍰ ⍰ Monitored data availability for faster query response by user.⍰ ⍰ Designed conceptual schema as per the client needs.⍰ ⍰ Looked after the security of company data from external access and threats.⍰ ⍰ Tested and installed latest versions of Database Management Systems in firm.⍰ ⍰ Granted access to users as per their needs and requirements.⍰ ⍰ Skills Used⍰ sql server management studio,visual studio Education bsc in computer science lagos state university - Lagos, GU Skills SQL, Microsoft PowerPoint, Windows XP, Microsoft office, Assembly Language, Microsoft Windows 7, C, C#, Visual basic and java programming ,Microprocessors ,Hardware/Systems Design, Digital Electronics, System organization, Microsoft excel.

Figure 2: Resumes Comparison: yellow: Pyresparser; green: Pre-trained NER; blue: Skill Ner; purple: Pre-trained NER+Skill Ner; dark blue: Pyresparser+Skill Ner; dark green: Pyresparser+Pre-trained NER; red: Pyresparser+Skill Ner+Pre-trained NER

of keywords in the database domain, albeit with less granularity compared to the former model.

3.2.3 Skills Matching

To find out a better way to match the resumes' skills with the job postings' skills, we conduct background research on similar topics. Bulut built a book recommendation system by computing TF-IDF and cosine similarity between features and books' descriptions which appeared to be effective (Bulut et al., 2018). Therefore, we also apply this method in matching the resumes and job postings for the recommendation system. Initially, we process the results from the 3 models related to resumes and the 2 models related to job postings. We compute the Term Frequency-Inverse Document Frequency (TF-IDF) metrics for the skill terms extracted from these models. Subsequently, we calculate the cosine similarity scores based on the TF-IDF metrics for skill terms in six scenarios, encompassing pairwise combinations between the 3 resume models and the 2 job posting models. Based on the similarity metrics, along with user-inputted resumes, 10 personalized job recommendations can be generated with the order of their similarity scores. In (figure 3), we input the resume 00063 (can be found in our github) and receive 10 personalized job recommendations, the figure shows only the first 5 jobs.

The cosine similarity between two vectors $\mathbf{A}$

	568416	595943	609284	574334	590389
0	0.569772	0.567944	0.545433	0.536776	0.53479

Figure 3: Recommendation System Output

and $\mathbf{B}$ is given by:

$$\text{Similarity}(\mathbf{A}, \mathbf{B}) = \frac{\sum_i a_i \cdot b_i}{\sqrt{\sum_i a_i^2} \cdot \sqrt{\sum_i b_i^2}}$$

4 Evaluation

4.1 Skills Extraction

Since the provided dataset lacks pre-labeled information for skill extraction, we decide to use manual annotation to add ground truth labels related to skill-related vocabulary to the data. A total of 100 samples are randomly selected from both the resume and job posting datasets. Each sample undergoes manual annotation by our team of four individuals, who carefully assigned B/I/O labels to every tokenized word, signifying their relevance to skills. The final label for each word is determined based on the majority consensus among the annotators. This approach ensure a comprehensive and reliable ground truth, capturing the nuances of skill-related vocabulary across diverse datasets. The incorporation of multiple perspectives through the collaborative annotation process enhances the robustness and accuracy of our evaluation framework. The remaining resumes and job postings constitute the training set. This meticulous partitioning allows for robust model evaluation and training on distinct subsets, ensuring a comprehensive and unbiased assessment of the recommendation system's performance.

4.1.1 Evaluation Metrics

In both manual and machine extraction, we apply a consistent approach to data cleansing and tokenization on the text to mitigate unnecessary variations at the word level. Despite our efforts, discrepancies between model-generated and manually extracted results are observed in terms of phrase length and word segmentation. For instance, the model may output "database management", while the manual extraction corresponds to "database management system" in the corpus. In our evaluation methodology, we do not favor considering these two extraction results as mismatched, as both "database management" and "database management system" imply the candidate's ability to manage database systems. However, due to objective differences in word length, we refrain from treating them as entirely equivalent. Therefore, we adopt a soft evaluation approach, specifically soft recall and precision (Fränti and Mariescu-Istodor, 2023).

In soft precision and recall, we avoid the binary definition of whether two phrases are equal (0 or 1) and instead quantify the degree of similarity between the two extracted phrases. This degree is a numerical value between 0 and 1(both included). To calculate it, we employ set theory. In set theory, cardinality is originally defined as the number of elements in a set. For the sets of ground truth values (G) and predicted values (P), we have

$$\text{card}(G \cap P) = \text{card}(G) + \text{card}(P) - \text{card}(G \cup P)$$

. Now we redefine cardinality with 'soft cardinality.' Firstly, we define the count of an element, say A, in a set G as the inverse of the sum of its similarity to all other items in the set, with the formula

$$\text{count}(A) = \frac{1}{\sum_{B \in G} \text{similarity}(A, B)}$$

. Here, the similarity between two strings is defined in relation to the Levenshtein distance by

$$\text{similarity}(A, B) = 1 - \frac{\text{edit}(A, B)}{\max(|A|, |B|)}$$

. Then, the soft cardinality of a set is defined as

$$\text{soft\_card}(G) = \sum_{A \in G} \text{count}(A)$$

, and the soft cardinality of the intersection is similar to the conventional cardinality of the intersection, defined as

$$\text{soft\_card}(G \cap P) = \text{soft\_card}(G) + \text{soft\_card}(P) - \text{soft\_card}(G \cup P)$$

With the definition of soft cardinality, we adapt soft precision and recall by replacing conventional cardinality with soft cardinality.

$$\text{Soft Precision} = \frac{\text{soft\_card}(G \cap P)}{\text{soft\_card}(P)}$$

$$\text{Soft Recall} = \frac{\text{soft\_card}(G \cap P)}{\text{soft\_card}(G)}$$

$$\text{Soft F-score} = \frac{2 \cdot \text{soft\_card}(G \cap P)}{\text{soft\_card}(G) + \text{soft\_card}(P)}$$

This modification provides a nuanced measurement of the similarity between two extracted phrases, considering their degrees of overlap.

	Precision	Recall	F1-score
Pyresparser	0.90853	0.72521	0.79537
pre-trained NER	0.98898	0.67650	0.79489
SkillNer_Resume	0.98072	0.64709	0.77481

Figure 4: precision, recall, and f1 score for three models

	Average Precision	Average Recall	Average F1-score
BIO+CRF	0.91196	0.79346	0.84147
SkillNer	0.91134	0.793905	0.84020

Figure 5: Job Posting Extraction Scores (Since the scores of the two models were very similar, the evaluation scores were kept to five decimal places to show the difference.)

4.1.2 Resume Evaluation Result and Analysis

Applying soft precision and recall, we evaluate the extraction results of three models designed for resume skill extraction. We calculate the average precision, recall, and F1 scores of the extraction results from each model on the same corpus set and compared them (figure 4).

It's noticeable that overall the precision is much higher than recall. Pyresparser has the highest f1-score and lower difference between precision and recall, while the other two models have a much higher difference between precision and recall. It indicates that Pyresparser is more suitable for scenarios where comprehensive coverage is crucial. The Pre-trained NER better fits the scenarios with a focus on minimizing false positives.

4.1.3 Job Posting Result and Analysis

In analyzing the current evaluation results (figure 5), it is observed that the precision, recall, and F1-score of the BIO + CRF model closely align with those of the SkillNer model. Notably, the BIO + CRF model exhibits a slightly enhanced precision compared to the SkillNer model, albeit to a minor extent.

4.2 Recommendation System

One hundred resumes from the validation set are randomly inputted, and four individuals independently observe the machine-generated matches. Scores assessing the appropriateness of the jobs recommended by six different combinations of machines (3 resumes skills extraction machines and 2 job postings skills extraction machines) are assigned to five of the jobs with the highest scores of similarity.

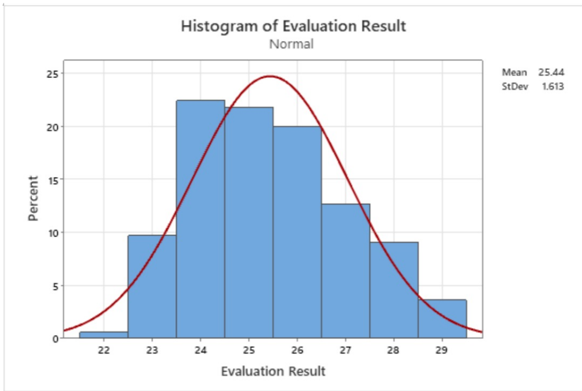


Figure 6: Observations Normal Plot

The general rule for giving the scores involves six aspects: key skills matching, additional skills matching, working experience matching, project experience matching, skills priority matching, and experience priority matching. To clarify, key skills refer to the skills starting with "required" or "must" in the job postings while additional skills refer to the skills starting with "preferred" or "bonus". Working experience is related to the job titles, together with the specific job descriptions in both the resumes and job postings. Similarly, project experience is associated with detailed project descriptions either at school or in previous work. Priority matching takes the ranking of the recommended jobs into account. The overall jobs should be in descending order of relevance. Every part accounts for 5 points (1 for 'not a match at all' and 5 for 'perfectly matched') and the total score is 30. This process allows us to calculate the appropriateness of the machine-generated recommendations.

After considerable manual scoring and cross-validation for 100 resumes over 6800 job postings, we get the final average scores with all the 6 pairs of methods.

Referring to the distribution plot (figure 6), we can see that the distribution has a shape similar to the normal distribution with a mean of 25.44 and a standard deviation of 1.613. It indicates that manual tagging can effectively distinguish between fair matchings and unfair ones. We do not show strong evidence of individual bias against the scoring.

Comparing the average scores for each combination of the machines, we draw to a conclusion of the best model combination with the highest score (Approximate to four decimal places):

As the table exhibits (table 1), overall all six machines have quite similar average scores. Among

Table 1: Recommendation System Scores

Resume Model	Job Model	Average Score
Pyresparser	BIO + CRF	25.6563
	SkillNer	25.2813
Pre-trained NER	BIO + CRF	25.4688
	SkillNer	25.4783
SkillNer	BIO + CRF	25.2609
	SkillNer	25.4348

them, the Pyresparser and BIO + CRF models reached the highest average score at 25.6563, which is close to the total score of 30, indicating the efficiency and accuracy of the job recommendation system. Therefore, we chose Pyresparser and BIO + CRF model as our final machine.

5 Discussion

In the evaluation process of both skills extraction and the recommendation system, there are aspects that require further discussion. One significant challenge lies in the manual evaluation of recommendations. Despite the establishment of a set of rules to guide and score the recommendations, the subjective nature of manual evaluation introduces the potential for bias. The process involves four different individuals scoring the same recommendation results, but the lack of certification raises concerns about the objectivity and appropriateness of the ground-truth data provided. However, obtaining truly ideal ground-truth data would involve tracking users' actual choices following system recommendations, which is time-consuming and requires a significant amount of effort.

Besides, due to the limited quantity of available job posting datasets, our evaluation results may not fully capture the efficiency and accuracy of our recommendation system. The suboptimal evaluation outcomes could stem not only from inherent issues within the recommendation system but also from the possibility that our dataset lacks suitable positions for recommendation.

6 Conclusion

We develop an efficient and accurate recommendation system where users input their resumes, and in return, receive a curated list of 10 job postings arranged in order of relevance.

We observe that the Pyresparser method yielded the best extraction results when comparing three different methods for extracting skills from resumes. When evaluating two methods for extracting skills from job postings, we find their effectiveness to be close. Upon integrating the results from resume and job posting extractions to establish our recommendation system, manual comparisons reveal that the combination of Pyresparser and the BIO + CRF method produces the most favorable outcomes. The result of recommendation system evaluation align with that of terminology extraction evaluation, specifically, the methodology that yielded the best results in terminology extraction also demonstrated superior performance in the evaluation of recommendation systems.

We make our code publicly available at: `https://github.com/gdgdandsz/NLP_job_recommendation_system`.

7 Future Work

In the future, we aim to enhance our recommendation system through practical implementation in real-world settings. We plan to gather additional users' feedback, encompassing metrics such as click-through rates and user evaluations for each recommended job posting after users input their resumes. Additionally, if feasible, we aspire to collect data on users' application rates for the recommended positions and the outcomes of these applications, including whether they are hired and their post-hire experiences. Besides, we plan to continuously gather a larger range of job positions to enhance the diversity of available job postings.

By leveraging this comprehensive users' feedback, we intend to continuously refine our recommendation algorithm, ensuring its adaptability and effectiveness in addressing the evolving needs of users in the job-seeking process. This iterative improvement process aligns with our commitment to providing a robust and user-centric recommendation system.

In addition, to make the system more user-friendly, a web page or a GUI system can be developed so that all the users have access to use the recommendation system with clear instructions. Moreover, a database can be set up with frequently updated data for job postings. In that way, users can have the most updated job information together with the recommendation results to prepare themselves better for job hunting.

References

Betül Bulut, Buket Kaya, Reda Alhajj, and Mehmet Kaya. 2018. A paper recommendation system based on user's research interests. *2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*.

Silvia Fareri, Nicola Melluso, and Gualtiero Fantoni Filippo Chiarello. 2021. Skillner: Mining and mapping soft skills from any text. *Expert Systems with Applications*, 184.

Pasi Fränti and Radu Mariescu-Istodor. 2023. Soft precision and recall. *Pattern Recognition Letters*, pages 115–121.

Thomas AF Green, Diana Maynard, and Chenghua Lin. 2022. Development of a benchmark corpus to support entity recognition in job descriptions. *Proceedings of the 13th Conference on Language Resources and Evaluation (LREC 2022)*, page 1201–1208.

K. F. F. Jiechieu and N. Tsopze. 2021. Skills prediction based on multi-label resume classification using cnn with model predictions explanation. *Neural Comput Applic*, pages 5069–5087.

Behrang Mohit. 2014. *Named Entity Recognition*, pages 221–245. Springer Berlin Heidelberg, Berlin, Heidelberg.

Mike Zhang, Kristian Nørgaard Jensen, and Barbara Plank Sif Dam Sonniks. 2022. Skillspan: Hard and soft skill extraction from english job postings. *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4962–4984.