Natural Language Processing

Project 1

Done By:

Alessandro Gentili

Marc Martinez

Manon Lainaud

Nigel Teo

Abstract

This project aims to develop a job recommendation system that efficiently matches resumes to job descriptions using a Two-Tower Embedding Model enhanced with contrastive learning, FAISS indexing, and cosine similarity. The system leverages Transformer-based embeddings (Sentence-BERT) and TF-IDF to generate high-dimensional vector representations of resumes and job descriptions. These are then processed through a deep learning-based twin-tower model, one tower encodes job descriptions and the other encodes resumes.

The model is trained using contrastive learning to maximize the similarity between relevant job-resume pairs while distinguishing irrelevant ones. To facilitate real-time job searching, FAISS (Facebook AI Similarity Search) is employed for fast approximate nearest neighbor retrieval. Additionally, cosine similarity is used to compute a percentage match score, providing a transparent evaluation of job-resume compatibility.

Users can query the system using either a job description or their own CV in text format, with results ranked based on their similarity scores. This approach ensures an efficient, scalable, and accurate job matching system that enhances candidate-job pairing in recruitment pipelines. Future improvements could include fine-tuning Transformer embeddings, incorporating domain-specific features, and exploring alternative loss functions for improved accuracy.

Credits

We would like to acknowledge the following individuals and resources for their contributions to this project:

\begin{itemize}

\item \textbf{Anna Wroblewska}, for his invaluable guidance and support throughout the project.

\item \textbf{The authors of the mentioned papers}, whose methodologies and insights were integral to shaping our approach.

\end{itemize}

We also wish to thank our peers and tutors for their constructive feedback and contributions to the project.

1. Introduction

Natural Language Processing (NLP) is a branch of artificial intelligence (AI) that enables computers to understand, analyze, and generate human language. It integrates techniques from computer science, linguistics, and statistical modeling to process large volumes of textual or spoken data. NLP has a wide range of applications, including chatbots, sentiment analysis, speech recognition, and information retrieval.

In the modern job market, the sheer volume of job postings and applications presents a significant challenge for both recruiters and job seekers. Traditional keyword-based matching systems often fail to capture the nuances of a candidate's skills, experience, and career trajectory, leading to suboptimal job recommendations. Such information retrieval systems primarily rely on exact term matching, but as Spärck Jones (1996) emphasizes, NLP's real value lies in its ability to extract and summarize meaningful information beyond individual words.

A major challenge in building effective job recommendation models is the lack of perfect ground truth labels. Unlike traditional supervised learning tasks with clearly defined correct answers, job-resume matching lacks a universal standard. Candidates may qualify for multiple roles based on diverse skills and experiences, while hiring decisions often depend on subjective factors that are difficult to quantify. A study done by Dong et al. (2024) highlights how typically, only items previously recommended to users have associated ground truth data, which complicates the evaluation and fairness assessment of such systems. This ambiguity makes it challenging to train classification models with explicit positive and negative labels.

To overcome this limitation, AI-driven job recommendation systems utilize natural language processing (NLP) and deep learning to derive meaningful job-resume relationships without relying on predefined labels. Contrastive learning techniques, such as twin-tower embedding models, enable the system to encode job descriptions and resumes into high-dimensional vector spaces, allowing for a semantic similarity assessment beyond simple keyword matching. By integrating Transformer-based embeddings (e.g., Sentence-BERT), TF-IDF features, and FAISS indexing, this project aims to develop a scalable and efficient job recommendation system that ranks candidates based on semantic relevance and cosine similarity scores. This project aims to develop an AI-driven job recommendation tool that can extract and analyze relevant

information from CVs, assisting human resource (HR) departments in candidate classification during initial recruitment phases. Additionally, the system will provide personalized job recommendations for candidates, helping them discover roles that align with their skills and experience.

NLP techniques will be used to identify and extract key candidate attributes, such as years of experience, relevant skills, and job location. However, for this project, we focus on three primary aspects:

1. Job Titles – Standardizing and matching relevant job roles.
2. Relevant Skills – Extracting and comparing required skills from resumes and job descriptions.
3. Full Descriptive Text – Analyzing and embedding complete resume/job descriptions to capture deeper contextual relevance.

These three aspects were chosen based on their significance in employer decision-making and their effectiveness in distinguishing candidates.

Job titles serve as the initial filtering mechanism in recruitment, helping employers quickly assess role suitability. Decorte et al. (2021) highlight the critical role of job titles in human resource processes, demonstrating that similar job descriptions often align with similar skill sets. Their study introduces a neural representation model that leverages job titles to enhance job-resume matching, reinforcing their importance in this project.

While job titles provide a high-level categorization, skills play a crucial role in determining an applicant's actual capabilities. Jiechieu and Tsopze (2020) emphasize the importance of skill extraction and prediction in resume processing. Their research highlights that identifying the key skills required for a job—often known in advance—enables more accurate candidate matching, making skill extraction essential for building effective job recommendation systems.

Beyond job titles and skills, full descriptive text offers a richer understanding of candidate-job fit. Liu et al. (2019) discuss the integration of full-text information from job postings and resumes, showing that a more context-aware approach improves job recommendations. Their

findings justify embedding complete job descriptions and resumes to capture deeper semantic relationships.

By leveraging NLP-driven semantic matching, this project aims to improve recruitment efficiency and job discovery, bridging the gap between job seekers and recruiters through AI-enhanced job matching.

In this document, we will begin by conducting a comprehensive literature review of state-of-the-art (SOTA) methods and tools in NLP, recommendation systems, and CV-JD matching. This review will explore existing research and methodologies, with a particular focus on comparison tables for datasets, methods, and pre-trained models. Following this, we will develop and document the solution concept for feature extraction from CVs and JDs, focusing on the project's approach and methodology. This will include a detailed explanation of the approach, covering methods for feature extraction, merging, and initial model selection.

2. Related work

The Two-Tower Embedding Model is inspired by advancements in recommendation systems that leverage contrastive learning and deep learning-based embeddings for scalable retrieval tasks. Several existing works have influenced the development of our system, particularly Uber's Two-Tower Embeddings Model for personalized recommendations and a TF-IDF and Cosine Similarity-based approach used in movie recommendation systems. By integrating insights from these methodologies, we designed a model that effectively captures semantic relationships between job descriptions and resumes, ensuring accurate and context-aware matching.

Uber's Two-Tower Embeddings Model (2023) demonstrates how contrastive learning can be applied in large-scale recommendation systems. Their model uses separate embedding towers to encode users and content (e.g., restaurants, drivers, or job postings) into a shared vector space, enabling efficient similarity searches using cosine similarity or nearest-neighbor retrieval. This framework allows the system to match user preferences with recommendations without requiring explicit feature engineering. Inspired by this, we adapted a Two-Tower Model for resume-job matching, where one tower processes job descriptions while the other processes resumes, ensuring that both embeddings are optimized to reflect meaningful job-resume relationships.

Additionally, the TF-IDF and Cosine Similarity-based Movie Recommendation System by Singh et al. (2020) provides a strong foundation for content-based filtering. This approach models text similarity using vector space representations, where each document is transformed into a feature vector, and the cosine of the angle between two vectors determines relevance. The advantage of using TF-IDF is its ability to assign greater importance to key terms while reducing the influence of commonly occurring words. By incorporating TF-IDF embeddings alongside transformer-based Sentence-BERT embeddings, we ensure that our model captures both keyword-driven and semantic similarities.

By merging these methodologies, our system benefits from the scalability of Two-Tower Embeddings, the semantic richness of transformer-based models, and the explainability of TF-IDF and Cosine Similarity. This hybrid approach enables our model to prioritize meaningful features in job descriptions and resumes, rather than relying solely on job titles. Furthermore, by

integrating FAISS for nearest-neighbor retrieval, our system efficiently retrieves high-dimensional embeddings, making it suitable for real-world recruitment applications.

This paper also introduces CONFIT, a contrastive learning-based framework designed to improve resume-job matching by leveraging data augmentation by Yu et al. (2024). The authors define the matching task as a function $f(R, J) \rightarrow R$, where a neural network determines the compatibility between a resume (R) and a job (J). The system is evaluated using two datasets.

One challenge in job-resume matching is data sparsity, as candidates apply to only a few jobs, limiting the availability of positive interaction data. CONFIT mitigates this issue through data augmentation, where parts of resumes (e.g., work experience) are paraphrased to create synthetic resumes while maintaining the same relevance labels. A similar augmentation process is applied to job descriptions, expanding the dataset without altering underlying relationships.

To further enhance match accuracy, contrastive learning is used to generate high-quality embedding representations. The system constructs training instances where each job is paired with a positive matching resume and a set of negative mismatched resumes. An encoder network is then trained to differentiate between suitable and unsuitable pairs, optimizing a cross-entropy loss function. Instead of manually encoding each resume/job field separately, CONFIT employs a simplified encoder architecture, where a linear layer fuses field representations into a single dense vector for efficient comparison. The final matching score is computed using an inner product similarity function.

CONFIT's scalability and ranking efficiency make it a robust system for real-world hiring applications. The framework achieves state-of-the-art performance, improving nDCG@10 scores by up to 19\% for job ranking and 31\% for resume ranking compared to prior methods. Additionally, despite not being explicitly optimized for classification, CONFIT remains competitive in classification tasks. The paper also discusses ethical considerations, particularly regarding bias mitigation and privacy protection, emphasizing the need for fairness in AI-driven recruitment systems.

Furthermore, Das et al. (2018) proposed an Entity Extraction Model for resume parsing using Natural Language Processing (NLP) techniques. Their work emphasizes the importance of Named Entity Recognition (NER) for extracting structured information from unstructured text,

particularly in resumes where formatting varies significantly. Their system applies common NLP techniques such as part-of-speech (POS) tagging, tokenization, and entity linking to improve the quality of extracted resume features. Given the diverse formats of resumes (e.g., plain text, XML, PDF), their method ensures robust entity extraction across multiple input types. Inspired by this work, we integrate NER techniques to extract essential skill keywords from raw job descriptions and resumes, improving the semantic matching process.

By merging these methodologies, our system benefits from the scalability of Two-Tower Embeddings, the semantic richness of transformer-based models, and the explainability of TF-IDF and Cosine Similarity. Our approach ensures that meaningful features in job descriptions and resumes are prioritized, rather than relying solely on job titles. Additionally, FAISS indexing allows for efficient retrieval of high-dimensional embeddings, making our system scalable and suitable for real-world recruitment applications.

3. Solution Concept & Approach

Now that we've done a little review of the Literature Review & Background, we are going to see the Solution Concept & Approach.

The proposed solution aims to optimize the matching between resumes (CVs) and job descriptions (JDs) through a structured approach. The first step involves data collection. The necessary datasets include CVs, which contain candidate profiles with information such as skills, experience, education, and job history, and JDs, which outline the job requirements, responsibilities, and desired skills.

Once the datasets are collected, the next phase involves data cleaning and preprocessing, which is critical for ensuring the accuracy and relevance of the analysis. This process begins with stopword removal, which eliminates common, non-informative words such as "am," "is," "are," "of," "a," and "the." These words do not contribute meaningful information to the underlying topics and are removed to enhance the dataset's semantic clarity. Next, non-English rows are discarded to ensure linguistic consistency across the dataset, which is essential for accurate natural language processing (NLP).

Following this, Named Entity Recognition (NER) is employed using SpaCy to extract meaningful entities from both CVs and job descriptions (JDs). Key focus areas for extraction include job titles, skills, certifications (e.g., programming languages, tools), as well as education and years of experience, which are crucial for aligning candidate qualifications with job requirements.

In addition, high-frequency words that appear in a significant percentage of documents (e.g., 80–90\%) are removed, as these terms are often non-distinct and fail to provide valuable differentiation between various job roles or qualifications. These preprocessing steps align with methodologies proposed by Jagwani et al. (2023), who demonstrate the effectiveness of combining entity detection using SpaCy's NER and Latent Dirichlet Allocation (LDA) to extract meaningful semantic representations for resume evaluation. Their approach focuses on creating a content-driven scoring system, achieving an overall accuracy of 82\\% by considering attributes such as education, work experience, and skills (Jagwani et al., 2023).

3.1 Two-Tower Embedding Model: Learning Semantic Similarity Through Contrastive Learning

At the heart of this system is the Two-Tower Embedding Model, which leverages contrastive learning to encode job descriptions and resumes into a high-dimensional vector space. Unlike traditional keyword-based matching systems, which rely on lexical similarity, contrastive learning techniques help the model understand deeper semantic relationships between job descriptions and candidate resumes.

Contrastive Learning for Job-Resume Matching

Contrastive learning is a self-supervised learning technique that trains a model to bring similar pairs closer while pushing dissimilar pairs apart in the embedding space. In this project:

- Positive pairs: A resume and a job description that are likely a good match.
- Negative pairs: Randomly selected job descriptions and resumes that are less relevant to each other.

Contrastive learning is particularly effective in representation learning. By training the model to distinguish between similar and dissimilar data pairs, it ensures that similar items are closely clustered in the embedding space while dissimilar ones are positioned further apart. This approach is also used in the methodology described in ConFit: Improving Resume-Job Matching using Data Augmentation and Contrastive Learning, where contrastive learning is leveraged to address data sparsity and improve matching accuracy. By embedding resumes and job descriptions into a shared vector space, the model enables efficient similarity measurement, achieving significant improvements in ranking performance over traditional approaches such as BM25 and neural embeddings. The use of inner product similarity to compute matching scores further demonstrates the effectiveness of this strategy, which serves as a basis for our implementation plan (Yu et al., 2024).

To improve the accuracy and relevance of resume-job matching, we will employ contrastive learning as a core technique for evaluating embeddings and a good way to determine similarity

measurement. This approach enables the system to learn dense embeddings for both CVs and JDs within a shared vector space, facilitating a more precise computation of matching scores. The inner product of these embeddings will then be used to calculate a matching score, ranking jobs for a CV by their matching percentage. This methodology ensures that resumes and job descriptions are not only compared on textual similarity but also on contextual alignment.

This contrastive approach ensures that the model does not rely on explicitly labeled positive and negative examples, which are often subjective and difficult to define in real-world hiring scenarios. Instead, it learns meaningful relationships directly from the data, improving generalization and robustness in job recommendations.

Twin-Tower Architecture

In order to maintain clarity and avoid confusion, CV and JD datasets will be processed separately. This independent preprocessing ensures that the unique attributes of each dataset are preserved, enabling the model to effectively differentiate between the qualifications presented in CVs and the requirements outlined in JDs. Specifically, CVs capture information such as experiences, skills, and educational qualifications, while JDs emphasize job-specific requirements, responsibilities, and desired expertise. By processing these datasets independently, the model can accurately represent the distinct characteristics of each, leading to more effective resume-job matching.

This methodology aligns with the approach described in *ConFit: Improving Resume-Job Matching using Data Augmentation and Contrastive Learning* by Yu et al. (2024). The authors emphasize the importance of representing CVs and JDs as dense embeddings to encapsulate their unique information and improve matching accuracy. Their work demonstrates that independent preprocessing and representation of these datasets enhance the system's ability to rank jobs and resumes effectively, achieving significant improvements in performance metrics such as nDCG@10 (Yu et al., 2024).

The Two-Tower Model consists of two independent deep learning networks:

- Job Tower: Encodes job descriptions and required skills.

- Resume Tower: Encodes candidate resumes and their listed skills.
- Similarity Computation: Both embeddings are mapped to a shared vector space, and their similarity is computed using cosine similarity.

By structuring the model this way, job descriptions and resumes can be embedded separately but comparably, enabling fast retrieval in large-scale hiring scenarios.

3.2 Enhancing Representation Learning: Transformer-Based Embeddings, TF-IDF, and FAISS

To improve the quality of resume and job embeddings, the system integrates multiple NLP-based feature representations, the first of which are transformer-based embeddings. We use Sentence-BERT (SBERT) to generate contextual embeddings for both job descriptions and resumes. Unlike traditional word embeddings (e.g., Word2Vec, GloVe), SBERT captures sentence-level meaning and provides robust representations that help differentiate similar but distinct job roles.

Term Frequency-Inverse Document Frequency (TF-IDF)

Our proposed approach also incorporates Term Frequency-Inverse Document Frequency (TF-IDF) as a key technique for feature extraction from resumes (CVs) and job descriptions (JDs). TF-IDF is widely used in information retrieval and natural language processing to identify and prioritize words or phrases that are not only frequent within a document but also distinctive across a corpus. By applying TF-IDF, the system ensures that terms specific to a resume or job description—such as technical skills, certifications, or unique job requirements—are given higher weight, while common words that appear across documents are deprioritized.

TF-IDF operates by calculating two key components:

1. Term Frequency (TF): This measures how often a term appears in a document relative to the total number of terms in that document.

2. Inverse Document Frequency (IDF): This scales the importance of a term inversely with its frequency across all documents in the corpus. Words that are common across many documents, such as "and" or "the," receive lower scores, while unique terms are weighted more heavily.

$$TF - IDF(t, d) = TF(t, d) \times IDF(t)$$

Where:

$$TF(t, d) = \frac{Number\ of\ occurrences\ of\ term\ t\ in\ document\ d}{Total\ terms\ in\ document\ d}$$

$$IDF(t) = log\left(\frac{Total\ number\ of\ documents}{Number\ of\ documents\ containing\ term\ t}\right)$$

By leveraging TF-IDF, the proposed system extracts key features from CVs and JDs that are particularly relevant to matching candidates to job postings. This approach aligns with the methodologies discussed in Das et al. (2018), where the authors emphasize the importance of feature extraction from unstructured data for better alignment with recruitment processes. The TF-IDF model not only captures domain-specific terminology but also reduces the influence of generic terms, enabling a more precise and meaningful representation of resumes and descriptions.

This feature extraction forms the foundation for subsequent processes in the pipeline, such as embedding generation and similarity computation, ensuring the system focuses on critical aspects of the resume-job matching task.

FAISS and Cosine Similarity for Final Ranking

To enable real-time job searching, we use FAISS (Facebook AI Similarity Search), an optimized nearest-neighbor search library. Once the model generates embeddings for job descriptions and

resumes, FAISS is able to index these embeddings in a high-dimensional space and retrieves the closest matching resumes efficiently, even in large datasets.

After FAISS retrieves the top-k matches, cosine similarity is applied to rank the final job-resume pairs. Since cosine similarity measures the angle between two vectors, it is ideal for comparing embedding representations. A higher cosine similarity score indicates a stronger match between a job and a resume.

Cosine similarity measures the angle between two vectors in an n-dimensional space. Instead of looking at the absolute values of the vectors, it focuses on their direction. It is fast to compute even for high-dimensional data and works well with sparse data (e.g., TF-IDF matrices)

Here is the formula for cosine similarity:

$$cos(\theta) = \frac{A \cdot B}{||A|| \times ||B||}$$

Where:

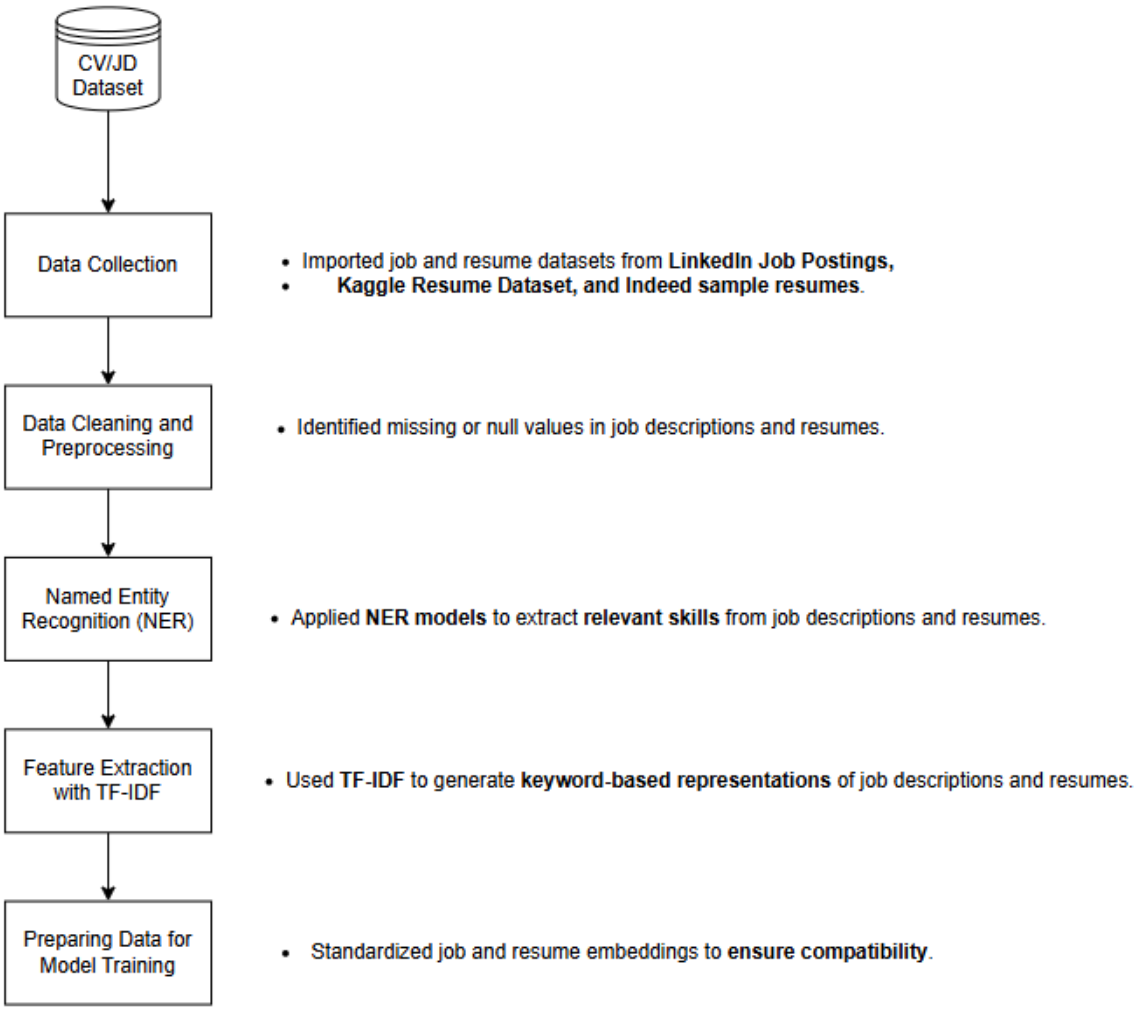
- A and B are two vectors (e.g., text embeddings or TF-IDF vectors).
- $A \cdot B$ is the dot product of the two vectors.
- $||A||,\ ||B||$ are the Euclidean norms (magnitudes) of the vectors
- θ is the angle between the two vectors

4. Experiments

In this experiment, we processed two datasets that share a similar structure, except for the resume data, where we retained the Category column to preserve job sector information. This preprocessing step ensures that both resumes and job descriptions are structured in a way that facilitates embedding-based similarity computations. Our goal was to create a clean, structured dataset that allows for efficient retrieval and ranking using transformer-based embeddings, TF-IDF, and FAISS.

For our experiments, we utilized two publicly available datasets that provide structured job descriptions and resumes: the LinkedIn Job Postings dataset and the Resume Dataset. These datasets serve as the foundation for our job-resume matching system, allowing us to evaluate how well our approach generalizes across diverse job roles and candidate profiles. While both datasets contain textual job-related information, they differ in structure and content, with the resume dataset containing an additional Category feature that helps preserve job sector classifications.

While both datasets provide valuable textual information for job-resume matching, they differ in structure. The LinkedIn Job Postings dataset is centered around job requirements, whereas the Resume Dataset focuses on candidate qualifications. The Category field in the resume dataset introduces an additional layer of information, allowing us to evaluate how well the model generalizes across industries. By processing these datasets with a unified approach—combining contrastive learning, Sentence-BERT embeddings, TF-IDF features, and FAISS indexing—we ensure that our system can effectively match job postings with relevant resumes while preserving meaningful job sector classifications.



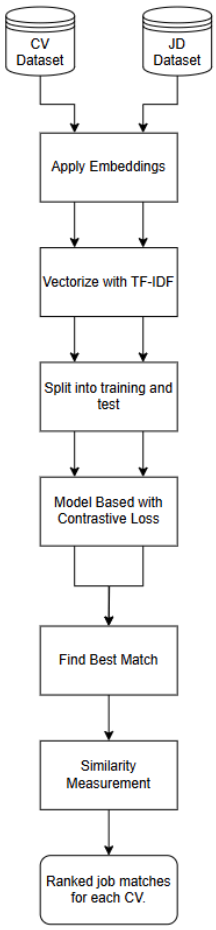
Our preprocessing pipeline began with standard text cleaning, including the removal of unnecessary characters, lowercasing, and tokenization to convert text into meaningful components. We also handled missing values appropriately to maintain data integrity. This is to prepare for the cleaned data to be transformed the text into vector representations using Sentence-BERT (SBERT) to generate contextual embeddings, ensuring that job descriptions and resumes were represented in a way that captures semantic meaning rather than relying solely on keyword-matching.

The resume dataset followed a similar process, using title, text, and skills, but with one key difference: we retained the Category column. Unlike the job dataset, which only contained raw job descriptions and skill requirements, the resume dataset-maintained job sector

classifications such as "HR," "Engineering," and "Healthcare." This additional feature allows for potential category-wise analysis, making it possible to evaluate whether job matches are biased toward specific industries or if the system provides diverse recommendations across different job sectors. The resumes were then converted into embeddings using SBERT and TF-IDF, and their skills were encoded in the same multi-hot format as the job dataset.

Retaining the resume category was crucial for evaluation. It allows us to conduct an industry-level breakdown of matching results, enabling us to assess whether job recommendations align well within each sector. Additionally, it improves interpretability by allowing category-wise ranking, which helps us analyze the system's performance in different job domains. This information is also useful for clustering analyses, such as t-SNE visualizations, to observe how job embeddings are distributed across different resume categories.

By structuring the dataset in this manner, we ensured that semantic similarity computations are both meaningful and efficient. The final processed output consists of job and resume embeddings indexed for fast retrieval, with an additional layer of industry classification retained for resumes to enable more fine-grained analysis. This structured approach allows us to accurately compare job descriptions and resumes while gaining deeper insights into category-specific job matching trends.



Twin Tower Architecture Model

Once the data was preprocessed, we constructed the Two-Tower Model, which consists of two independent deep learning networks—one for job descriptions and another for resumes. Each tower processes textual inputs using Sentence-BERT embeddings, which generate high-dimensional vector representations that capture semantic meaning rather than relying solely on keyword overlaps. These embeddings were further passed through dense layers and normalized to ensure compatibility in the shared embedding space. The model was trained using contrastive loss, optimizing it to minimize the distance between positive job-resume pairs while maximizing the separation of negative pairs.

To efficiently retrieve the most relevant resumes for a given job description, we leveraged FAISS (Facebook AI Similarity Search), an indexing system optimized for high-dimensional similarity search. After training, the resume embeddings were indexed within FAISS, allowing for

near-instantaneous retrieval of the closest matching resumes. When a new job description was provided, the system computed its vector representation and performed a nearest-neighbor search within the indexed resume database. To refine the ranking of retrieved resumes, we employed cosine similarity, ensuring that resumes with the highest semantic alignment were prioritized.

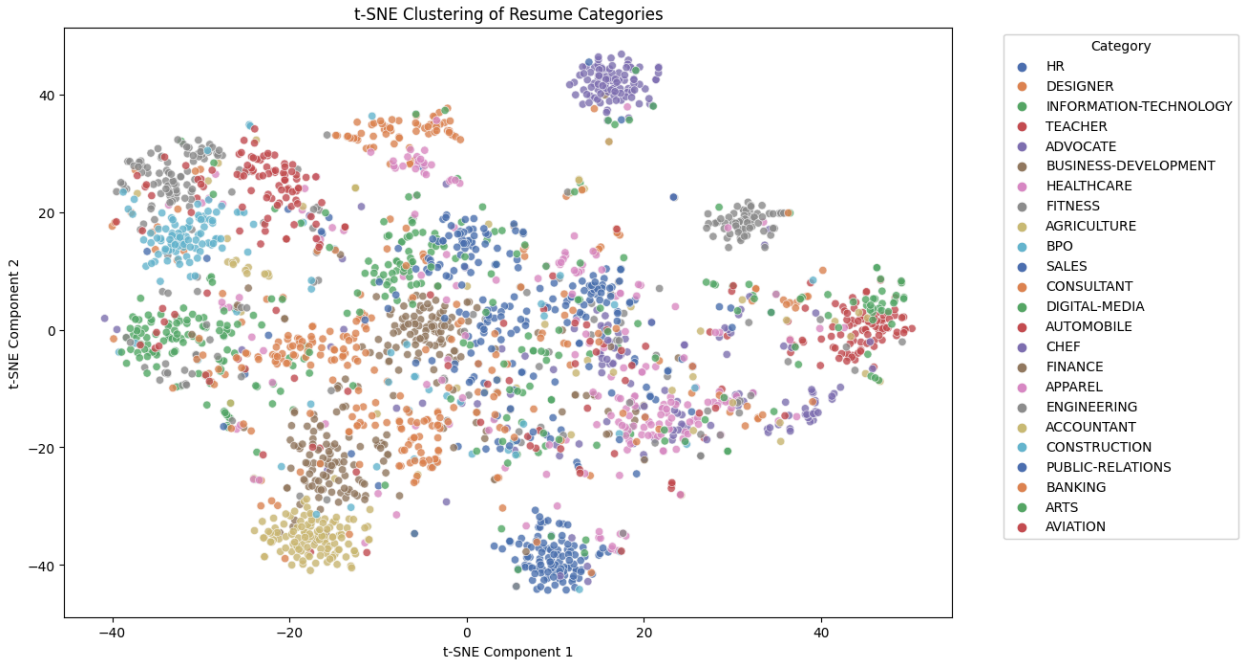
Experiment Design and Synthetic Resume Generation

One of the primary challenges in testing resume-matching models is the ethical concerns surrounding external resume usage. Real-world resumes often contain sensitive personal information, including names, contact details, work history, and education records. Using such resumes for model evaluation without explicit consent poses significant privacy and data security risks. Additionally, compliance with data protection laws such as GDPR (General Data Protection Regulation) and PDPA (Personal Data Protection Act) necessitates careful handling of personally identifiable information (PII). Given these concerns, we had to find an alternative approach that maintained ethical integrity while still allowing us to test the model in realistic conditions.

To address these challenges, we opted to use template resume samples from Indeed to test our model. These templates provide structured and anonymized resume data that resemble real-world resumes without containing sensitive personal details. By leveraging standardized resume samples, we ensured that our dataset remained free of private information while still representing diverse job sectors, experience levels, and skills. This approach allowed us to maintain data privacy and compliance while evaluating how well our model generalizes across different job roles.

By incorporating both standardized Indeed resume templates, we ensured that our evaluation process was ethical, diverse, and representative of real-world job applications. This approach not only eliminated privacy concerns but also provided a controlled testing environment where we could systematically assess our model's performance across different industries and career trajectories.

5. Analysis



This t-SNE visualization represents the clustering of resume embeddings into different job categories after applying Sentence-BERT (SBERT) embeddings.

The t-SNE plot reveals distinct clusters for certain job categories, indicating that the embedding model successfully captures meaningful patterns within resume texts. For example, certain well-defined clusters can be observed for HR, Information Technology, Healthcare, and Aviation, suggesting that resumes from these fields share similar language structures, skills, and contextual meanings. The grouping of these categories implies that Sentence-BERT embeddings effectively differentiate resumes based on their domain-specific vocabulary and context.

The visualization also highlights some isolated clusters, such as those in the top-right and bottom-left corners of the plot. These could indicate specialized resumes that differ significantly from the majority, potentially representing niche job roles or unique skill sets. The presence of these isolated clusters suggests that the embedding model can distinguish highly specific

resume types, though further analysis is needed to determine whether these represent true distinctions or noise in the data.

Upon testing the model on various resume samples from Indeed, the job recommendation model reveals several key insights regarding its effectiveness in matching resumes with job descriptions, the results of which can be found in **Appendix A**. One of the most striking observations is that the FAISS evaluation proves to be of no use, as its results are often entirely mismatched, with match percentages as low as 0\% or within single digits. This outcome makes sense, as the model was trained using cosine similarity rather than relying on FAISS to determine closeness, which naturally results in low scores. However, while a lower confidence score is expected, FAISS should still provide sensible matches, yet it fails to do so, further indicating that it does not contribute meaningfully to job-resume matching and should likely be replaced with a more effective retrieval method.

Anomalies were also identified in specific job roles, such as the iOS Developer resume, which was unexpectedly matched first with a Manager position and also with a Corporate Engineering Support Technician. This suggests that the model struggles with highly specialized roles, possibly due to the way embeddings prioritize broader engineering terms rather than platform-specific expertise. However, there are also cases where the model performs well in recognizing specialization, such as the Java Developer resume, which correctly matches first with a Java Intern position. This indicates that the model can distinguish between general and specialized programming roles.

Despite these issues, the model demonstrates strength in recognizing job relevance beyond just job titles. It does not merely match roles based on superficial similarities but considers the underlying job descriptions, which is a positive indication of its effectiveness. This strength is particularly evident in the healthcare sector, where nurses are correctly matched with nursing-related roles, and Physical Therapists are accurately paired with Physical Therapist Technicians, indicating that the model successfully understands industry hierarchies and sub-specializations.

A notable strength in the model is its handling of law-related job resumes. The Legal Assistant resume often matches with roles such as Cashier, Sales, and Customer Service, however, this makes sense because the resume lacks substantial examples of legal experience or law-related terminology. When evaluating the Legal Secretary, the model performs much better, with Legal

Assistant appearing as the top match. This suggests that the model does understand law-related roles to some extent, but the dataset likely lacks sufficient law-related job descriptions, affecting the accuracy of legal job matches.

Additionally, the model shows consistency in matching certain professions. Teaching resumes align well with teaching jobs, and therapists match correctly with therapist-related roles, reinforcing the model's ability to recognize professional categories accurately. However, inconsistencies arise in roles like Warehouse Delivery Driver, where the highest matches are for Sales and Warehouse jobs, while Automobile Transporter and Mover appear much lower in ranking. This discrepancy is likely due to insufficient training data, suggesting a need to expand the dataset with more specific job descriptions for these roles.

To improve the model's performance, several steps can be taken. First, expanding the dataset to include a broader and more representative sample of job descriptions, particularly in underrepresented fields such as law and specialized technology roles, would enhance its accuracy. Additionally, ensuring that resumes contain more domain-specific terminology would help in better contextual matching. Finally, given the poor performance of FAISS, it would be beneficial to explore alternative retrieval methods that provide more meaningful results. Fine-tuning embeddings to better capture domain-specific expertise would also be a valuable step toward refining the model's accuracy.

In summary, while the model effectively captures industry-relevant information and does not simply rely on job titles, it still faces significant limitations in certain domains. The primary areas for improvement include addressing dataset imbalances, enhancing legal job descriptions, and removing ineffective evaluation methods like FAISS. With these refinements, the model could provide more precise and reliable job recommendations across all industries.

6. Conclusion and Future Work

In this document, we explored the potential of Natural Language Processing (NLP) to optimize the recruitment process by developing a system that efficiently matches resumes (CVs) with job descriptions (JDs). The project aimed to address key challenges in recruitment, such as unstructured data, large-scale datasets, and the need for accurate candidate-job matching, by leveraging state-of-the-art NLP techniques and methodologies.

Through a comprehensive literature review, we examined existing works and methodologies, including Named Entity Recognition (NER), Term Frequency-Inverse Document Frequency (TF-IDF), and contrastive learning, which informed the foundation of our proposed solution. This approach integrates multiple preprocessing techniques, feature extraction methods, and advanced machine learning algorithms to process CVs and JDs independently while preserving their unique attributes.

Our proposed system demonstrates the ability to extract critical information such as job titles, skills, certifications, and years of experience from CVs and JDs, using tools like SpaCy and TF-IDF. The solution further enhances the matching process by generating dense embeddings through contrastive learning, enabling precise similarity computations and ranked job recommendations for each candidate.

By combining insights from the literature with a structured, modular approach, the project offers a scalable, automated solution that reduces the time and effort required in the recruitment process for both candidates and human resources teams. Additionally, the methodology ensures a fairer and more efficient recruitment process by aligning candidate profiles with job requirements based on contextual and semantic relevance.

This work highlights the importance of NLP in transforming recruitment and sets the stage for further advancements, such as incorporating multilingual datasets, improving bias mitigation, and integrating more dynamic models for real-time job matching. As a result, the project not only contributes to improving recruitment efficiency but also paves the way for innovative applications of NLP in human resources and beyond.

References

Bhawal, S. (n.d.). Resume Dataset. Kaggle: Your Machine Learning and Data Science Community. https://www.kaggle.com/datasets/snehaanbhawal/resume-dataset?select=data

Das, P., Pandey, M., & Rautaray, S. (2018, September). A CV Parser Model using Entity Extraction Process and Big Data Tools. ResearchGate. https://dx.doi.org/10.5815/ijitcs.2018.09.03

Decorte, J., Van Hautte, J., Demeester, T., & Develder, C. (n.d.). JobBERT: Understanding job titles through skills. arXiv.org. https://doi.org/10.48550/arXiv.2109.09605

Dong, Y., Jin, K., Hu, X., & Liu, Y. (n.d.). Measuring fairness in large-scale recommendation systems with missing labels. arXiv.org. https://doi.org/10.48550/arXiv.2406.05247

Douze, M., Guzhva, A., Deng, C., Johnson, J., Szilvasy, G., Mazaré, P., Lomeli, M., Hosseini, L., & Jégou, H. (n.d.). The Faiss library. arXiv.org. https://arxiv.org/abs/2401.08281

Jiechieu, K. F., & Tsopze, N. (2020, August 28). Skills prediction based on multi-label resume classification using CNN with model predictions explanation. SpringerLink. https://link.springer.com/article/10.1007/s00521-020-05302-x

Koneru, A., & Yu, Z. (n.d.). LinkedIn Job Postings (2023 - 2024). Kaggle: Your Machine Learning and Data Science Community. https://www.kaggle.com/datasets/arshkon/linkedin-job-postings

Lewis, D. D., & Spärck Jones, K. (1996, January 1). Natural language processing for information retrieval. Communications of the ACM. https://doi.org/10.1145/234173.234210

Ling, B. (2023, July 26). Innovative recommendation applications using two tower embeddings at Uber. Uber Blog. https://www.uber.com/en-PL/blog/innovative-recommendation-applications-using-two-tower-embeddings/

Liu, M., Wang, J., Abdelfatah, K., & Korayem, M. (2019, July 23). Tripartite vector representations for better job recommendation. arXiv.org. https://doi.org/10.48550/arXiv.1907.12379

Resume Samples and Examples To Inspire Your Next Application. (n.d.). Indeed. https://www.indeed.com/career-advice/resume-samples

Singh, R. H., Maurya, S., Tripathi, T., Narula, T., & Srivastav, G. (2020, June). Movie Recommendation System using Cosine Similarity and KNN. https://dx.doi.org/10.35940/ijeat.E9666.069520

Yu, X., Zhang, J., & Yu, Z. (2024, October 8). ConFit: Improving resume-job matching using data augmentation and contrastive learning. ACM Conferences. https://dl.acm.org/doi/abs/10.1145/3640457.3688108

Appendix A

Type of Resume	Cosine Similarity Evaluation	FAISS Evaluation
Accountant	- staff accountant (Match: 76.85\%) - accountant (Match: 76.78\%) - accountant (Match: 75.51\%) - senior accountant (Match: 75.09\%) - accountant (Match: 74.94\%)	- business development (Match: 8.64\%) - forms designer (Match: 8.38\%) - accountant (Match: 3.37\%) - hr generalist (Match: 1.11\%) - consultant (Match: 0.00\%)
Branch Manager	- branch manager (Match: 71.67\%) - branch banker (Match: 69.57\%) - business development specialist branch manager (Match: 69.44\%) - finance manager (Match: 69.15\%)	- consultant (Match: 12.01\%) - consultant (Match: 5.49\%) - finance manager (Match: 5.08\%) - accountant (Match: 0.27\%) - education officer senior education officer guidance counseling unit (Match:

	- branch banking coordinator (Match: 68.85\%)	0.00\%)
Bartender	- bartender (Match: 77.70\%) - ws bartender call (Match: 70.46\%) - bartender server trainer banquet event captain (Match: 70.22\%) - sous chef (Match: 68.92\%) - executive chef (Match: 67.23\%)	- intern (Match: 31.95\%) - senior graphic designer (Match: 9.87\%) - director information technology (Match: 9.13\%) - director information technology (Match: 5.15\%) - owner operator (Match: 0.00\%)
Chef	- lead chef food truck manager (Match: 79.12\%) - executive chef (Match: 78.24\%) - chef owner (Match: 77.89\%) - chef de cuisine (Match: 77.23\%) - food preparation workers grill chef (Match: 77.12\%)	- noc engineer (Match: 8.81\%) - finance accountant (Match: 4.96\%) - assistant store manager operations human resources (Match: 3.43\%) - hr manager (Match: 0.08\%) - graphic designer (Match: 0.00\%)

Construction Worker	- construction worker (Match: 73.17\%) - construction manager (Match: 72.06\%) - construction manager ii (Match: 71.56\%) - construction foreman (Match: 71.44\%) - senior construction manager (Match: 71.18\%)	- sales representative (Match: 9.57\%) - executive chef (Match: 4.14\%) - director client services films operations technical services west coast (Match: 3.74\%) - sales manager (Match: 2.88\%) - sales associate (Match: 0.00\%)
DevOps Engineer	- information technology specialist (Match: 68.96\%) - information technology specialist (Match: 68.75\%) - information technology specialist (Match: 68.44\%) - manager information technology building automation systems (Match: 67.97\%) - information technology coordinator (Match: 67.67\%)	- senior digital marketing specialist (Match: 4.19\%) - wms consultant (Match: 2.61\%) - director preschool teacher (Match: 1.14\%) - senior advisor national fundraising director (Match: 0.22\%) - vzw customer tech advocate (Match: 0.00\%)
Digital Marketing Specialist	- digital marketing account	- contract senior associate media planner sapientnitro

	manager (Match: 73.79\%) - digital marketing specialist (Match: 72.86\%) - digital marketing specialist (Match: 72.22\%) - digital marketing manager (Match: 72.21\%) - digital marketing specialist (Match: 71.47\%)	(Match: 8.60\%) - hr manager (Match: 5.37\%) - accountant (Match: 3.14\%) - sr digital analytics manager (Match: 3.08\%) - assistant manager h (Match: 0.00\%)
Electrical Engineer	- engineering technician (Match: 74.44\%) - multi skilled engineering manager (Match: 73.22\%) - electrical engineer (Match: 71.19\%) - engineering assistant (Match: 70.48\%) - engineering intern (Match: 69.76\%)	- staff accountant (Match: 16.92\%) - director president minturn fitness center (Match: 6.35\%) - hr coordinator (Match: 1.72\%) - sales associate (Match: 1.17\%) - programme finance associate (Match: 0.00\%)
Financial Adviser	- financial analyst intern (Match: 71.76\%) - finance manager (Match: 69.99\%)	- program support assistant (Match: 10.83\%) - construction installer (Match: 1.97\%)

	- management consultant (Match: 68.95\%) - financial editor assistant (Match: 68.59\%) - finance manager (Match: 68.48\%)	- education officer senior education officer guidance counseling unit (Match: 0.48\%) - executive chef (Match: 0.44\%) - digital project manager (Match: 0.00\%)
Hotel Manager	- accommodation service executive (Match: 69.99\%) - director public relations partnerships (Match: 69.66\%) - reservations agent front desk agent guest services agent pbx operator (Match: 69.19\%) - vp finance (Match: 68.48\%) - manager administration facilities (Match: 67.13\%)	- engineering intern (Match: 8.70\%) - team lead senior analyst (Match: 7.72\%) - director finance (Match: 6.27\%) - lead piping designer (Match: 4.13\%) - operations finance director (Match: 0.00\%)
HR Specialist	- hr manager (Match: 76.40\%) - senior hr business partner (Match: 75.40\%) - hr manager (Match:	- business solution project manager (Match: 7.77\%) - senior product development manager (Match: 3.12\%) - dining services coordinator

	75.21\%) - hr generalist (Match: 74.80\%) - sr hr generalist (Match: 74.53\%)	(Match: 1.53\%) - consultant (Match: 0.46\%) - consultant (Match: 0.00\%)
iOS Developer	- manager (Match: 66.48\%) - information technology intern test automation engineer (Match: 64.49\%) - corporate engineering support technician (Match: 63.51\%) - director engineering (Match: 63.35\%) - platform architect healthcare incubation lab hil (Match: 63.28\%)	- mineralogy engineering intern (Match: 3.64\%) - director quality improvement network facilitation (Match: 1.00\%) - digital marketing associate (Match: 0.35\%) - accountant (Match: 0.00\%)
Java Developer	- java intern (Match: 72.23\%) - information technology intern test automation engineer (Match: 71.77\%) - technical designer (Match: 70.03\%)	- sales representative (Match: 12.63\%) - mineralogy engineering intern (Match: 11.18\%) - associate teacher (Match: 5.70\%)

	- construction worker (Match: 69.86\%) - consultant (Match: 69.14\%)	- executive chef (Match: 5.15\%) - accountant (Match: 0.00\%)
Legal Assistant	- cashier (Match: 66.93\%) - customer service manager (Match: 66.44\%) - sales associate cashier (Match: 66.42\%) - customer service advocate (Match: 66.04\%) - sales associate (Match: 65.93\%)	- graphic designer (Match: 1.74\%) - senior accountant (Match: 1.70\%) - healthcare consultant (Match: 0.56\%) - chef (Match: 0.25\%) - digital marketing director (Match: 0.00\%)
Legal Secretary	- legal assistant (Match: 72.51\%) - consultant (Match: 69.45\%) - assistant company secretary (Match: 68.85\%) - owner attorney mediator (Match: 68.47\%) - consultant (Match: 67.97\%)	- business development specialist (Match: 15.15\%) - administrative assistant (Match: 5.70\%) - volunteer advocate (Match: 3.85\%) - information technology specialist (Match: 1.12\%) - engineering manager (Match: 0.00\%)

Marketing	- product marketing manager (Match: 71.95\%) - business development executive (Match: 71.41\%) - digital marketing manager (Match: 71.37\%) - digital marketing specialist (Match: 71.33\%) - business development rep (Match: 70.79\%)	- general manager (Match: 4.98\%) - hr manager (Match: 2.87\%) - sales (Match: 1.04\%) - accountant (Match: 0.56\%) - sales representative (Match: 0.00\%)
Medical Assistant	- licensed practical nurse step unit (Match: 71.92\%) - registered nurse (Match: 71.21\%) - healthcare provider (Match: 70.75\%) - registered nurse (Match: 70.74\%) - patient care technician (Match: 70.36\%)	- digital marketing manager (Match: 3.19\%) - freelance consultant (Match: 1.61\%) - business development manager (Match: 1.58\%) - manager (Match: 0.11\%) - associate manager desig (Match: 0.00\%)
Nurse Practitioner	- practicum experience (Match: 74.69\%)	- volunteer hr ivolunteer (Match: 10.60\%)

	- charge nurse (Match: 74.24\%) - licensed practical nurse step unit (Match: 73.90\%) - registered nurse (Match: 73.76\%) - field nurse (Match: 73.51\%)	- accountant (Match: 9.00\%) - digital marketing manager (Match: 7.32\%) - associate manager desig (Match: 6.41\%) - partner business development (Match: 0.00\%)
Physical Therapist Assistant	- physical therapist technician (Match: 73.97\%) - rehabilitation specialist massage therapist (Match: 70.72\%) - certified personal trainer (Match: 70.25\%) - dance educator (Match: 69.88\%) - physical therapy aide (Match: 69.73\%)	- digital strategy manager (Match: 17.15\%) - online coaching personal training (Match: 6.81\%) - substitute para professional (Match: 4.14\%) - digital marketing account manager (Match: 1.84\%) - accountant helper (Match: 0.00\%)
Product Manager	- consultant (Match: 70.54\%) - senior director product	- hr representative (Match: 7.43\%) - head accounts finance

	management (Match: 70.48\%) - information technology consultant managing member (Match: 69.53\%) - consultant (Match: 68.81\%) - director engineering (Match: 68.79\%)	(Match: 1.57\%) - level critical platform support engineer (Match: 0.39\%) - general manager (Match: 0.02\%) - accountant (Match: 0.00\%)
Quality Assurance Engineer	- qa engineering team lead (Match: 78.13\%) - qa test analyst (Match: 74.15\%) - qa engineering manager (Match: 73.31\%) - test analyst intern contractor (Match: 72.01\%) - adjunct instructor (Match: 71.61\%)	- business consultant (Match: 22.95\%) - technology specialist (Match: 8.96\%) - business development leader (Match: 6.00\%) - finance director (Match: 1.41\%) - sales manager (Match: 0.00\%)
Relationship Manager	- business development director (Match: 69.70\%) - business development	- sales associate (Match: 3.93\%) - sales associate (Match:

	manager (Match: 69.66\%) - consultant (Match: 68.84\%) - consultant (Match: 68.66\%) - operations manager (Match: 68.51\%)	3.81\%) - consultant (Match: 0.62\%) - business development manager (Match: 0.15\%) - consultant (Match: 0.00\%)
Sales Specialist	- sales (Match: 74.78\%) - sales consultant (Match: 74.40\%) - js sales representative psr (Match: 74.26\%) - sales manager (Match: 74.14\%) - sales representative sales management (Match: 73.81\%)	- teacher (Match: 5.74\%) - overnight pharmacy technician (Match: 4.35\%) - general manager executive chef (Match: 3.35\%) - operations manager (Match: 2.13\%) - kindergarten teacher (Match: 0.00\%)
Software Developer	- software engineering co op (Match: 69.20\%) - website designer (Match: 68.49\%) - corporate engineering	- engineering manager (Match: 5.90\%) - sr manager (Match: 3.72\%) - director quality improvement network

	support technician (Match: 67.92\%) - software engineering manager (Match: 67.85\%) - ea information technology specialist iii drupal dev (Match: 67.56\%)	facilitation (Match: 1.99\%) - product designer (Match: 0.88\%) - consultant (Match: 0.00\%)
Teacher	- ep high school english language arts teacher (Match: 76.56\%) - inclusion teacher (Match: 75.00\%) - teacher (Match: 74.55\%) - teacher (Match: 74.28\%) - bilingual language arts sixth grade teacher (Match: 74.03\%)	- engineering technologist (Match: 10.13\%) - accountant (Match: 4.14\%) - audit recovery specialist (Match: 1.40\%) - hr specialist (Match: 0.42\%) - director business development (Match: 0.00\%)
Therapist	- co founder therapist teaching artist (Match: 72.86\%) - court appointed special advocate abused neglected	- construction helper (Match: 13.48\%) - manufacturing technician ops coordinator (Match: 2.58\%)

	children (Match: 69.44\%) - dance educator (Match: 69.22\%) - family community advocate (Match: 68.65\%) - bilingual domestic violence advocate (Match: 68.63\%)	- substitute para professional (Match: 1.89\%) - operations research analyst (Match: 0.06\%) - staff accountant (Match: 0.00\%)
Warehouse Delivery Driver	- sales (Match: 70.08\%) - warehouse lead (Match: 68.81\%) - grocery clerk (Match: 68.26\%) - automobile transporter (Match: 68.08\%) - mover (Match: 67.88\%)	- finance officer (Match: 5.06\%) - sales associate (Match: 3.41\%) - group exercise fitness instructor (Match: 2.85\%) - chief digital officer (Match: 2.23\%) - customer care representative (Match: 0.00\%)

Appendix B

Paper Title	Techniques/Tools	Objectives	Challenges
Resume Parser Using Natural Language Processing Techniques	OCR, NLP techniques (lexical analysis, syntactic analysis, semantic analysis, NER), ElasticSearch dashboards	Streamline recruitment by parsing, structuring, and ranking resumes based on job requirements.	Handling biases, integrating social media data, and ensuring efficient information extraction.
A CV Parser Model using Entity Extraction Process and Big Data Tools	NER, entity relation extraction, tokenization, POS tagging, Hadoop MapReduce	Extract entities from resumes in varied formats and optimize processing with big data tools.	Managing context-dependent meanings of words and handling diverse data formats.
CONFIT: Improving Resume-Job Matching using Data Augmentation and Contrastive Learning	Data augmentation (EDA, ChatGPT), contrastive learning, embedding ranking	Enhance resume-job matching by addressing label sparsity and improving embedding quality.	Overcoming dataset sparsity, mitigating biases, and ensuring ethical use of data.