Talent Analytics for AI Resume Screening

BIA 660 Final Project

Group Members: Harshil Cherukuri, Ratan Chowdary Paruchuri, Sidharth Peri

Section I: Motivation and Research Question

In the modern day job market, Artificial Intelligence has begun to play a significant role in the application process for both employers seeking to attract talent, and individuals looking to discover new opportunities in their desired fields. Artificial intelligence is seen as a necessity to automate many tasks in the application process due to the sheer increase in volume of job applicants, and because of its ability to perform a large amount of tasks in an efficient manner, comparatively to the traditional "human" approach. An example of a use of AI systems is with resume screening. For most initial resume screenings in a job application process, AI systems are entrusted in scanning resumes and looking for keywords or phrases or qualifications that an applicant's resume has that presents itself as a good match for the requirements and responsibilities of the role. Therefore, it is becoming increasingly important to understand these AI systems in order for job applicants to tailor their resumes to ensure they pass through and in a way "solve" these AI systems for initial screenings. It is also important for employers to ensure they are attracting the top talent for their roles. An applicant usually looks at the job description and responsibilities when choosing to apply for a new job. Therefore it is imperative for employers to have the correct keywords and phrases in their descriptions that maximizes the applicant talent they will receive.

The main motivation behind this project is to create a tool that can provide this information to both job applicants and employers. Our goal is to essentially finetune an aspect of

talent analytics by helping applicants improve their resumes, and employers improve their job descriptions. We break down our research into two specific questions:

- **Research Question 1:** What are the linguistic characteristics of the best resumes for specific jobs and fields? Are there common language/keywords/phrases that make certain resumes stand out and show a high degree of similarity with the job requirements? How can a candidate improve their resume in order to crack a specific field they are interested in? Can we reverse this analysis to find the common language/keywords/phrases that make specific jobs stand out and show a high degree of matching with a majority of resumes?
- **Research Question 2:** Can a system used to match jobs on the market with resumes provide a meaningful and correct relationship between a candidate's ideal field of choice and their best fit jobs? Does a candidate's resume accurately reflect their own field of interest?

We believe our work can help provide job candidates with a holistic review of their resumes by simulating the AI matching systems used by the majority of industry companies, and give candidates and employers an edge to tailor their resumes/job descriptions to progress through recruiting cycles more efficiently and attract the best talent to their roles.

Section II: Background and Related Work

The main background research we conducted was understanding the type of AI systems currently used by employers. Most employers use either an ATS (Applicant Tracking Software) or AI Resume screening. Part of ATS deals with reviewing candidate information with keyword searches to identify those who most closely match job listing requirements. AI Resume screening

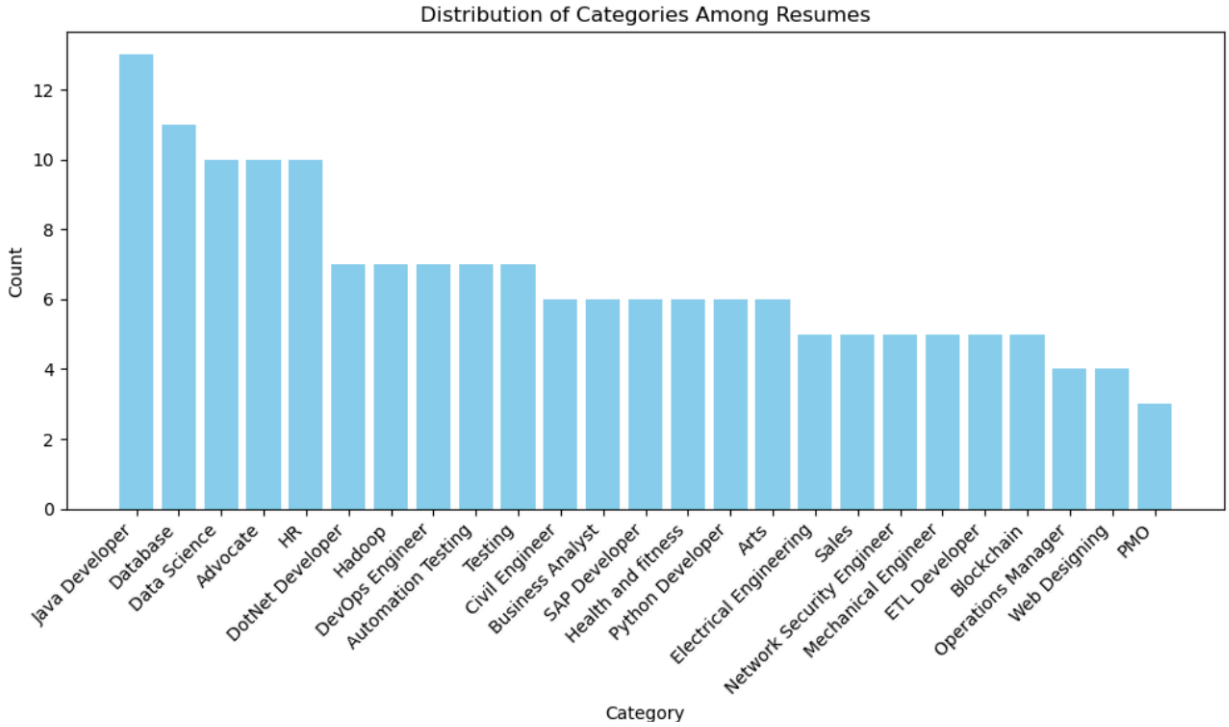
is also keyword-based, looking for specific phrases and patterns of text that match closely with the job listing requirements. The commonality between these two systems is the use of a similar keyword matching system that looks to identify and highlight resumes that have a high degree of matching with the job requirements. Therefore, we will look to mimic a similar system to understand and analyze the specific keyword characteristics for a high degree of matching. The paper, *A Comprehensive Survey of Artificial Intelligence Techniques for Talent Analytics* [1], touches on aspects of talent-related decisions in a quantitative manner. They perform analysis on Person-Job Fitting, a task that focuses on "measuring the matching degree between a job posting, consisting of job duties and job requirements, and a candidate's resume, consisting of work and educational experiences". Their work uses deep neural networks such as CNN and RNN to extract representation vectors from job postings and resumes and evaluate their fit through similarity measurements. Their implementations of extracting word vectors mainly consist of deep neural network based methods such as CNN, LSTM, and attention-based models. We look to achieve a similar representation using non deep learning methods, rather methods discussed in class such as pre-trained word embeddings, and TF-IDF vectors.

Section III: Methodology

Part 1: Generating Datasets

In order to implement our analysis, we needed to obtain a dataset of resumes and a dataset of job postings. For our resume data, we used a pre-scraped dataset found on Kaggle. This dataset consisted of approximately 200 scraped resumes, each having a specific resume category (the intended field for the resume) and a scraped description consisting of information such as educational background, skills, experiences, and career goals/motivations. Examples of

resume categories are: Java Developer, Database, Data Science, Business Analyst, and DevOps Engineer. The following graph shows the distribution among the different resume categories in our dataset.



For the jobs data, we manually web-scraped 120 job postings from Indeed using python's Selenium and BeautifulSoup libraries. Half of our job postings were targeted for data science and the other half were targeted for information technology. The information scraped was the job title and the main job description including educational qualifications, skill requirements, and responsibilities.

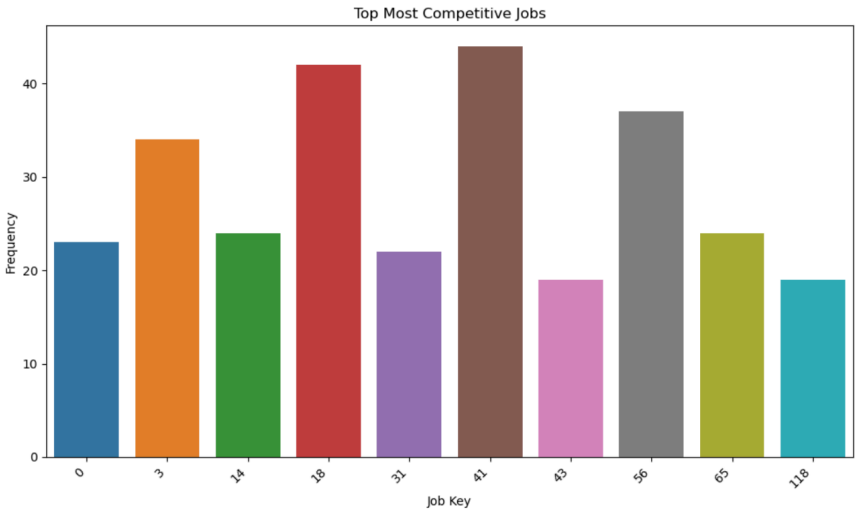
Part 2: Text Preprocessing and Cleaning

The next step was to perform initial preprocessing and cleaning of the textual data in the resumes and jobs datasets. For preprocessing, we removed any numerical digits, removed stop

words, added stemming and lemmatization, and made all text lowercase. These cleaned descriptions are now what we will use in our similarity models and analysis.

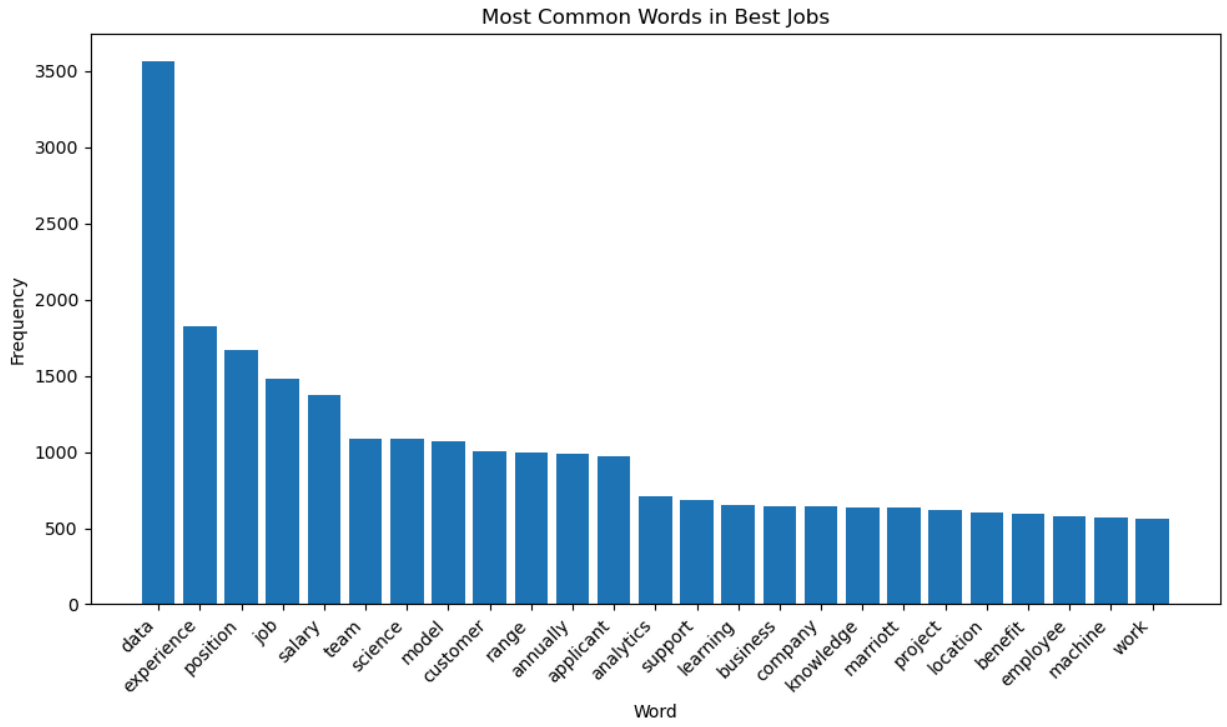
Part 3: Job Similarity Model

To address our first research question, we have created similarity models to generate a vector representation of the cleaned resume and job descriptions and determine their similarity. We have used two different methods of generating the vector representations. The first method was to use pre-trained word embeddings from GloVe and the second method was to generate word vectors using TF-IDF on the resume and job description corpus. After generating the word vectors for each resume and each job, we calculated the cosine similarity between each (resume, job) pairs. We created key variables (starting at 0) for each resume and job in order to keep track of the similarities and match jobs and resumes. For the job similarity model, we determined the jobs with the largest 5 cosine similarity values with each resume. The purpose of this model was to understand and analyze the most competitive and matched jobs. By analyzing their descriptions, we can see what linguistic characteristics and keywords make a job highly desirable/matchable in an AI system. After finding the 5 most similar jobs to each resume, we aggregated these results to all resumes and found what the 10 most matched jobs were. The following were the results for the GloVe embeddings model:



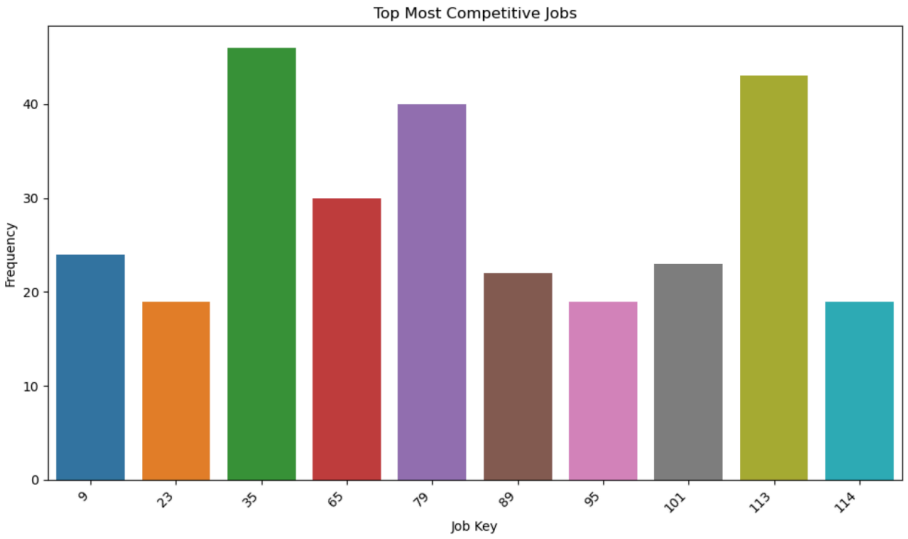
Job Key	Title
0	Data Scientist
3	Data Scientist (Remote Eligible)
14	Data Scientist
18	Data Scientist
31	Data Scientist
41	Data Scientist
43	Data Scientist
56	Data Scientist
65	Information Systems Specialist
118	System Administrator IV

For the GloVe embeddings model, the most matched job was job 41 which was a Data Scientist role. From these top ranked jobs, we wanted to determine the linguistic characteristics among them. Therefore, we looked at the distribution of the 25 most occurring words among these top ranked jobs. Seeing this distribution will allow us to analyze what keywords create a high matchability for jobs. The following is the distribution from the top jobs of the GloVe embeddings model.



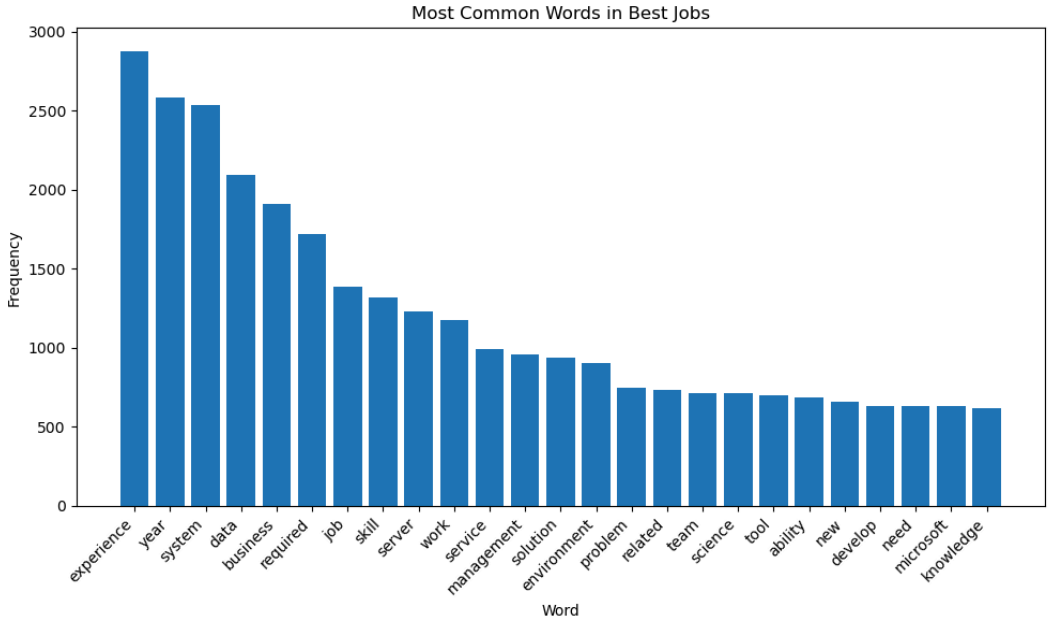
These were the 25 most occurring words among the top 10 ranked jobs and their frequencies. We can see that the most occurring word by far was “data”, followed by “experience”, “position”, “job”, and “salary.” Something to note among this distribution is that many of the words are generic, related to more corporate aspects of jobs such as the pay rate, experience needed, and type of position rather than skill related which would potentially be things like coding languages, and specific skills needed for the position.

The following are the same results using the model with TF-IDF vectors:



Job Key	Title
9	Data Scientist
23	Data Scientist
35	Data Scientists Analysts
65	Information Systems Specialist
79	System Administrator (virtual; remote; current...
89	PLEX ERP Analyst
95	Hybrid Systems Administrator
101	Systems / Network Administration
113	Systems Engineer
114	L3 Systems-Network Support Engineer

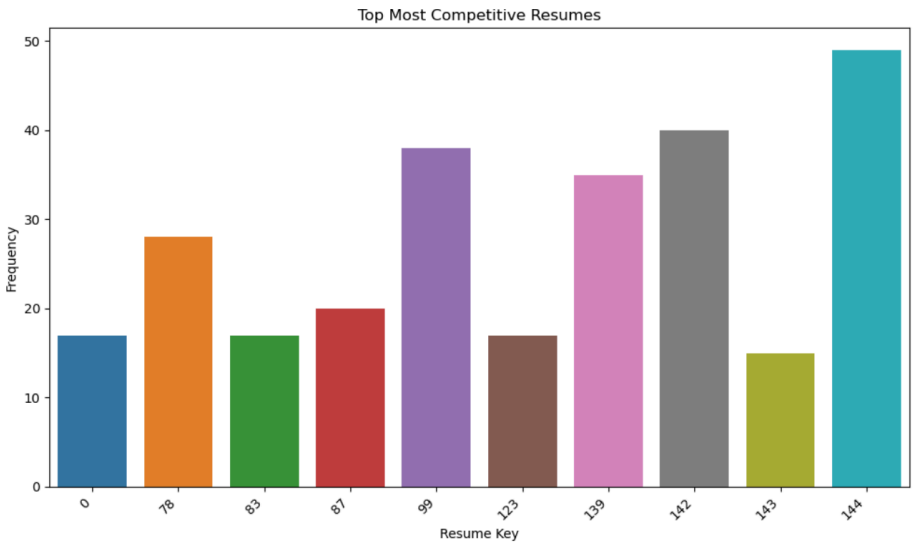
Looking at the top 10 ranked jobs from the TF-IDF model, an interesting observation is that there is no overlap between the GloVe word embeddings model and TF-IDF model. Another interesting thing to note is that the majority of the top ranked jobs for the TF-IDF model are information technology and systems jobs, whereas the majority of the top ranked jobs for the GloVe model were data science jobs. The distribution of the 25 most occurring words among these top jobs is the following:



Similar to the results from the GloVe embeddings model, we see that many of the top words are more corporate-based rather than skill-based. The top words from this model are words like: “experience”, “year”, “system”, and “data.” We also see some directional words such as “required” which may indicate that the model recognizes jobs that show a skill or educational requirement are coveted and well matched given the clarity they exhibit for their qualifications.

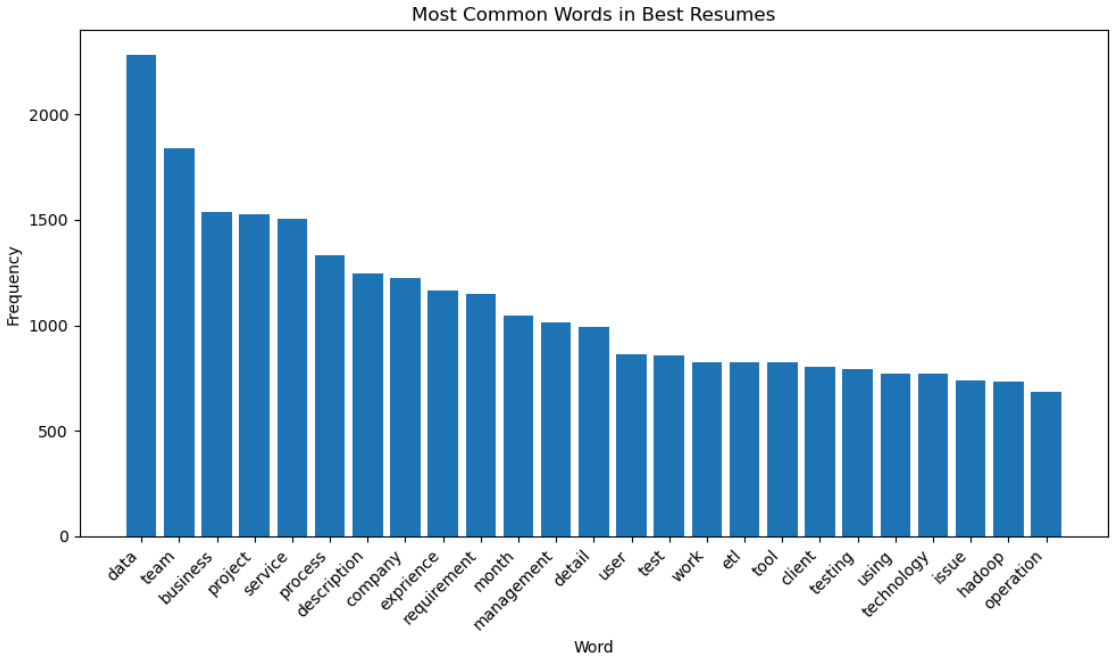
Part 4: Resume Similarity Model

We now have created a similar model as our job similarity model, but the purpose of this model is to find the top-matched or best resumes. For each of the 120 jobs, we find the 5 most similar resumes using cosine similarity for both the GloVe embeddings and TF-IDF models. Out of the most matched resumes, we again find the 10 most matched resumes, or the best ranked resumes, and the distribution of the 25 most occurring words among these resumes. The goal of this is to see what linguistic characteristics and keywords occur in the best matching resumes. These are the top ranked resumes using the GloVe embeddings model:

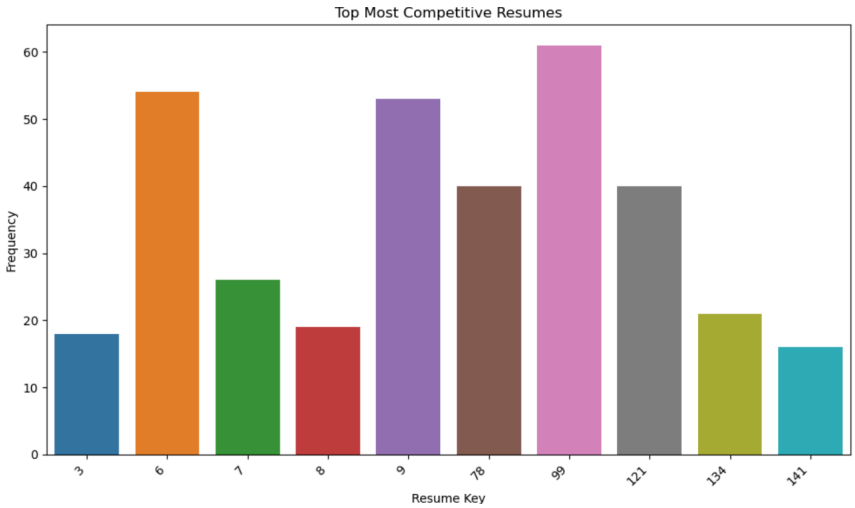


Resume Key	Category
0	Data Science
78	Business Analyst
83	SAP Developer
87	Automation Testing
99	Operations Manager
123	PMO
139	Hadoop
142	ETL Developer
143	ETL Developer
144	ETL Developer

From this we can see that the best resume, which results in the most similarity matches with the jobs dataset is resume 144 which has the category ETL Developer. The 25 most occurring words from these resumes has the following distribution:

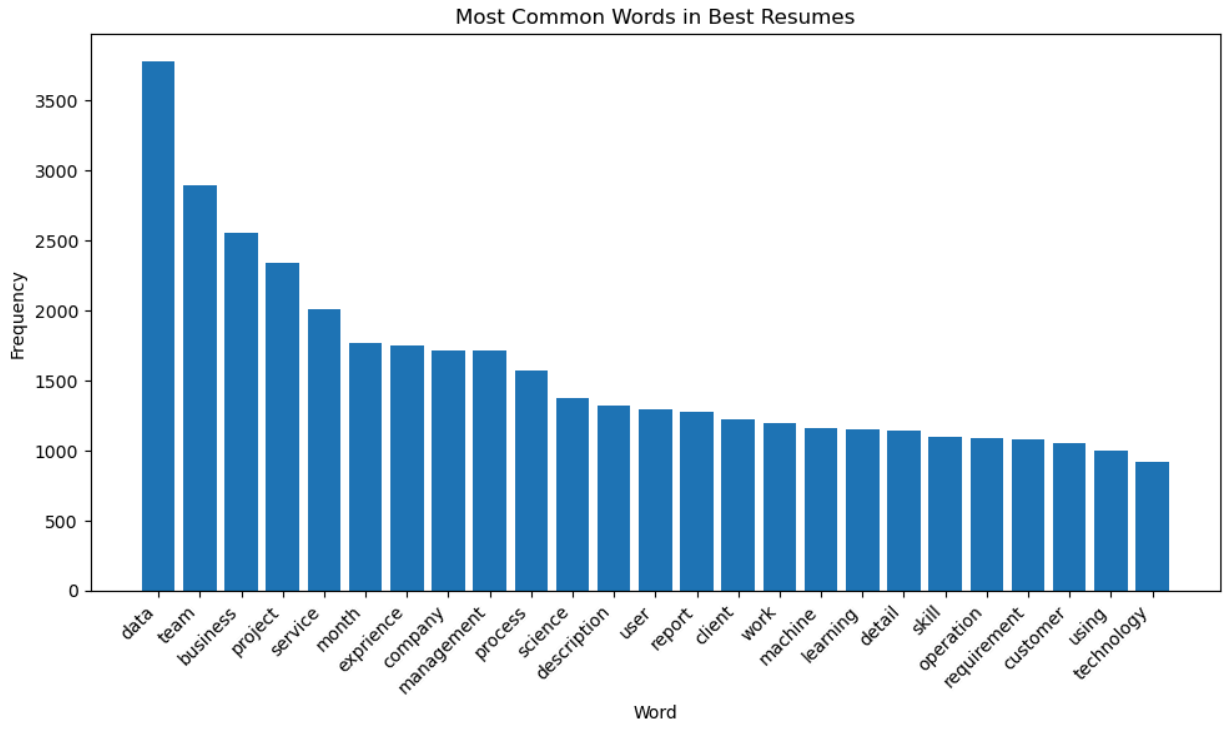


From the most common words in the top resumes, we see that there are more skill-based words among the top resumes. Words like “data”, “detail”, “test”, “etl”, “technology” and “hadoop” are among the most common. This shows that for keyword matching AI systems, having a resume with some skill-words can lead to a high degree of similarity with job requirement listings. We performed the same analysis using the TF-IDF model and obtained the following results for the top resumes:



Resume Key	Category
3	Data Science
6	Data Science
7	Data Science
8	Data Science
9	Data Science
78	Business Analyst
99	Operations Manager
121	PMO
134	Database
141	Hadoop

Comparing the results from the TF-IDF model and GloVe model, we can see that there were a few overlapping resumes, namely resume 99 and resume 78. For the TF-IDF model, resume 99 had the most matches. It is interesting to note that resume 99 which had the most matches is for the category Operations Manager, however the job titles are split between Data Science and Information Technology. This may suggest that having specific keywords on your resume may create a "false sense of similarity", meaning that a candidate who may not be intending for that job may coincidentally match. The following graph shows the distribution among the 25 most frequently occurring words:



From this graph, we can see that there is some significant overlap between the most common words in the top resumes amongst both models. We see high occurring words that could be potential bigram phrases, with words like: "data" and "science", and "machine" and "learning." These are highly sought after skills that would likely appear as some of the most popular

occurring phrases in the top resumes, so it makes sense that as individual words they appear as well.

Part 5: Resume Category Predictive Models

Our next goal was to address our second research question: Can a system used to match jobs on the market with resumes provide a meaningful and correct relationship between a candidate's ideal field of choice and their best fit jobs? Does a candidate's resume accurately reflect their own field of interest? Our method of determining this answer was to create 2 different classification models to predict the resume category field using the resume description, top matched job description, top matched job title, and similarity score as feature variables. The idea behind this model was to determine whether we can use an applicant's resume description and best matching job to obtain an accurate prediction for their target field or resume category. This model is not necessarily designed to maximize accuracy. A highly accurate model would reveal that the applicants clearly understand the strengths of their resume and our model category predictions match the resume categories. If we are able to build a strong model, we should be able to evaluate misclassifications and prove why our model, using the features we have outlined, would predict a resume to match a specific resume category that the applicant did not list as the primary category. This would be useful in fine-tuning a resume or potentially realizing different fields your skillsets are suitable for based on the AI matching systems applied in early resume screening rounds.

For obtaining a feature set, we first created a dataframe of each resume, its cleaned description, its top-matched job description and job title, and the similarity score (according to GloVe embeddings). In order to encode these textual attributes into numerical features, we

created TF-IDF matrices for the corpus of resume and job vocabulary and used the TF-IDF matrix as the encoded feature for our dataset. We also applied a TF-IDF vectorizer to the job titles column. These TF-IDF matrices were used as the main features in our classification models.

	aaa	aag	aakruti	aave	ab	abacus	abad	abap	abasaheb	abb	...	tech	technician	technology	texas	to	university	validation	virtual	year	Sim
0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.9
1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.9
2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.9
3	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.9
4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.9

This is an example of a subset of the final feature data used in the classification models.

For the 2 classification models, we have chosen to use a Random Forest Classifier, and a Support Vector Machine (SVM) model. While Random Forest is known for its robustness and ability to handle complex datasets, Support Vector Machine (SVM) is favored for its ability to capture intricate decision boundaries and generalize well to unseen data. SVM is particularly effective in high-dimensional spaces and datasets with clear margin of separation between classes. It works well for both linearly separable and non-linearly separable data by using different kernel functions to map the input features into higher-dimensional space. However, SVM can be computationally expensive, especially for large datasets, and may require careful selection of hyperparameters for optimal performance. Random Forest can capture complex nonlinear relationships between features and target variables. For our purposes, we implemented both models to see which gets better results as far as accuracy. After performing grid search to tune the model hyperparameters, we predicted resume category (a categorical variable with 25 different categories) on unseen, held-out resume data.

The Random Forest classifier had an accuracy of approximately 41% with the following being a subset of the results of the predictions:

Resume Key	Actual Category	Predicted Category
132	Database	Database
40	Mechanical Engineer	Operations Manager
157	Blockchain	Java Developer
115	DevOps Engineer	Testing
56	Civil Engineer	Civil Engineer
117	Network Security Engineer	Network Security Engineer
123	PMO	Java Developer
83	SAP Developer	SAP Developer
17	HR	HR
102	Operations Manager	DevOps Engineer
54	Health and fitness	Java Developer
143	ETL Developer	ETL Developer
150	DotNet Developer	Java Developer
24	Advocate	Advocate
148	DotNet Developer	Sales
4	Data Science	Advocate
139	Hadoop	Hadoop
37	Web Designing	Java Developer

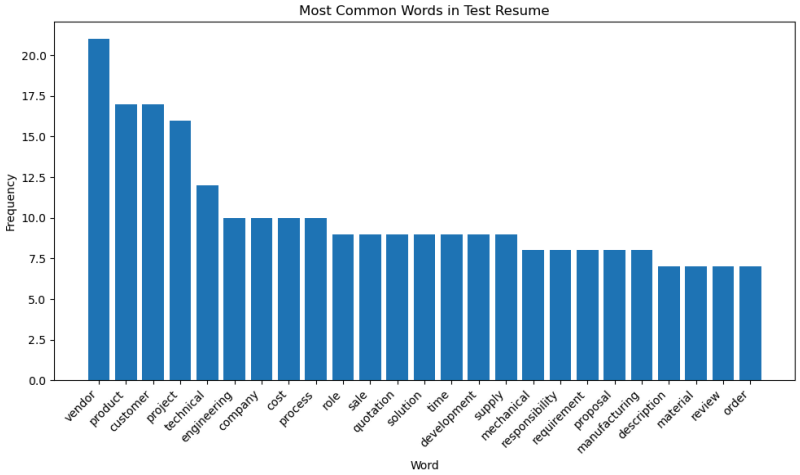
Considering that we are performing a multiclass classification on 25 different classes and that the goal of the model was not necessarily to maximize accuracy, we were very satisfied with our Random Forest results. We obtained many correct predictions, proving the accuracy of our model, but we also obtained some misclassifications which would allow us to analyze the linguistic features of the test resume and the collection of resumes of the predicted and actual target values to determine why the resume was misclassified and whether our model classification makes more sense. However, our results were not as good for the Support Vector Machine classifier as the accuracy was only around 12% and the predictions are as shown below:

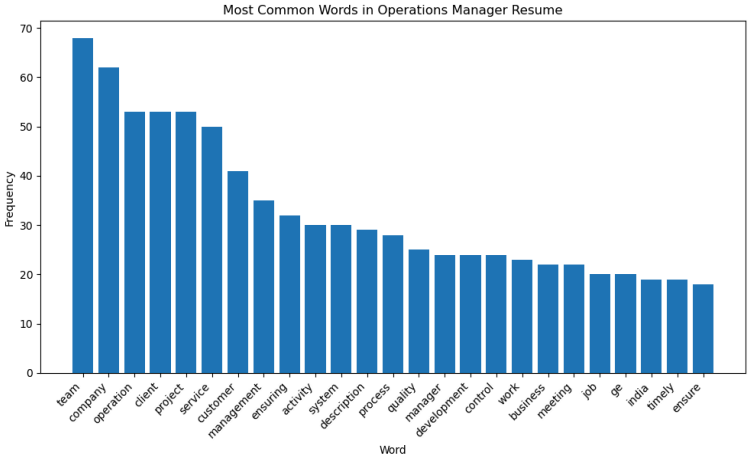
Resume Key	Actual Category	Predicted Category
132	Database	Arts
40	Mechanical Engineer	Business Analyst
157	Blockchain	DevOps Engineer
115	DevOps Engineer	Testing
56	Civil Engineer	Java Developer
117	Network Security Engineer	Java Developer
123	PMO	Java Developer
83	SAP Developer	Java Developer
17	HR	Database
102	Operations Manager	Java Developer
54	Health and fitness	Java Developer
143	ETL Developer	Business Analyst
150	DotNet Developer	Java Developer
24	Advocate	Java Developer
148	DotNet Developer	Database
4	Data Science	Data Science
139	Hadoop	Data Science
37	Web Designing	Java Developer

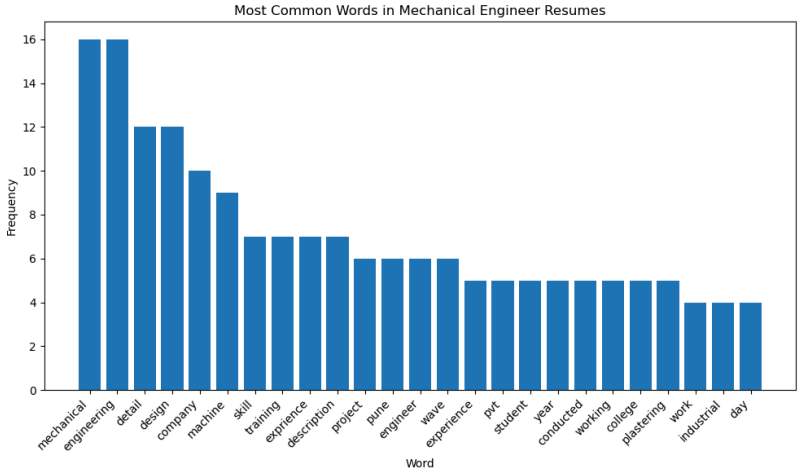
This model performs much worse than the Random Forest Classifier. This is likely due to the features being too complex for the capabilities of an SVM in comparison to an ensemble model like a Random Forest classifier. Therefore, in evaluating the misclassifications of our model we will use the Random Forest predictions.

In order to show why our model may have made specific misclassifications, we have looked at two specific misclassifications from our Random Forest model. We look specifically at the most occurring words within the test resume, and the most occurring words within the collection of resumes of their actual resume category (not including that resume) and the most occurring words within the collection of the predicted category resume. It is likely that there is a high degree of overlap between the common words from the test resume and predicted category resumes, leading to the misclassification. This could show the individual that based on their

skills they may fit better to other careers, or that they need to update and fix their resume to better match jobs they are targeting. The first misclassification we looked at was resume 40. Resume 40 was predicted by the Random Forest classifier to be an Operations Manager resume, while it actually was a Mechanical Engineer resume. The following graphs show the distribution of the most commonly occurring words among resume 40, operations manager resumes, and mechanical engineer resumes.



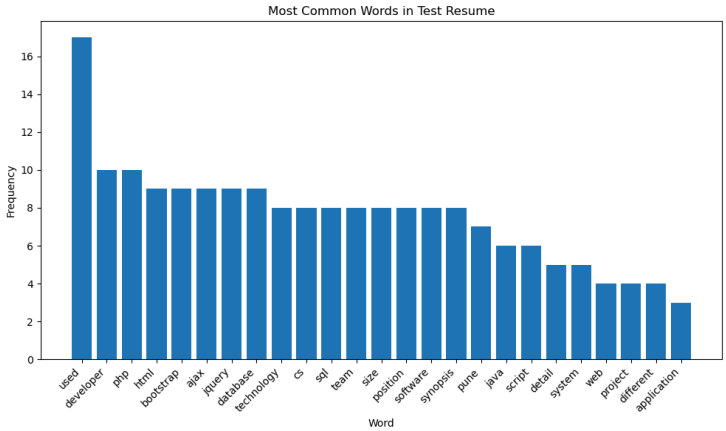


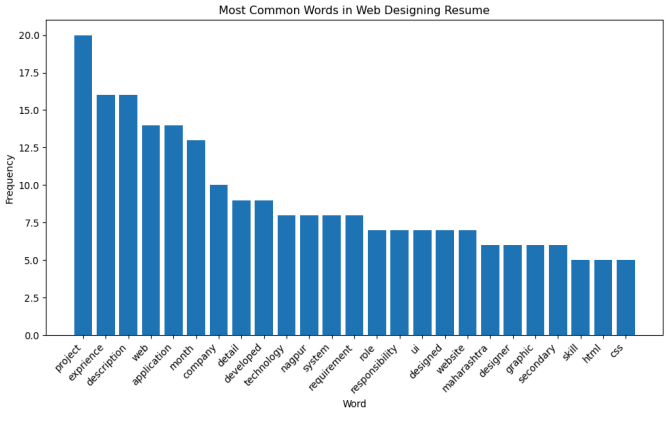


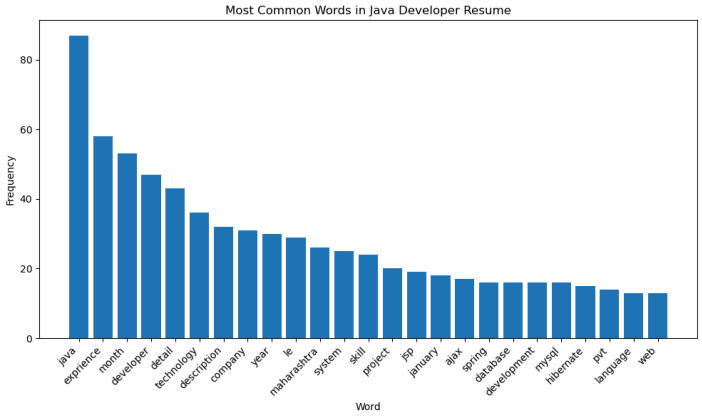
From these charts, we can see that there is some overlap between the frequently occurring in the test resume, and both the Mechanical Engineer and Operations Manager resumes. There are 6 common words with Operations Manager resumes: “company”, “customer”, “description”, “development”, “process”, and “project”. However, in the actual category for this resume,

Mechanical Engineer, there are only 5 common words: “company”, “description”, “engineering”, “mechanical”, and “project”. Therefore, it makes sense that our model predicted Operations Manager for this resume’s category. The two most common words (“mechanical” and “engineering”) in the Mechanical Engineering resumes both appear in the test resume, but the two most common words (“vendor” and “product”) from the Operations Manager resumes don’t appear in the test resume. Therefore, the amount of common words may be less important, rather the actual common words themselves are more pivotal in classifying a resume. This may be something lacking in our model that results in the resume being classified as Operations Manager, as Operations Manager and Mechanical Engineer are pretty different fields that probably would not have much overlap.

Another misclassification we would like to highlight is resume 37, which was a Web Designing resume that our model classified as Java Developer. This example is interesting because web designing and java developer are similar fields that likely have transferable skills. Therefore, our model making that misclassification might be sensible. The following graphs show the most frequently occurring words among the test resume, web designing resumes, and java developer resumes.







There are 9 overlapping words in the test resume and java developer resumes, which are: “ajax”, “database”, “detail”, “developer”, “java”, “project”, “system”, “technology”, and “web”. In the actual resume category, Web Designing, there are only 7 overlapping words: “application”, “detail”, “html”, “project”, “system”, “technology”, and “web”. We can see that some specific skills that the test resume has like: “ajax”, “database”, and “java” are skills that java developers likely need. Therefore, we think that our classification makes sense and may actually be more appropriate for this specific resume than their intended category which was Web Designing.

Part 6: Vector Databases Model

We want to develop a system that leverages the latest vector database technology for semantic search to enhance the job matching process. By analyzing job postings and comparing their semantic content with information from resumes, the system can accurately identify the most appropriate job titles for individual candidates.

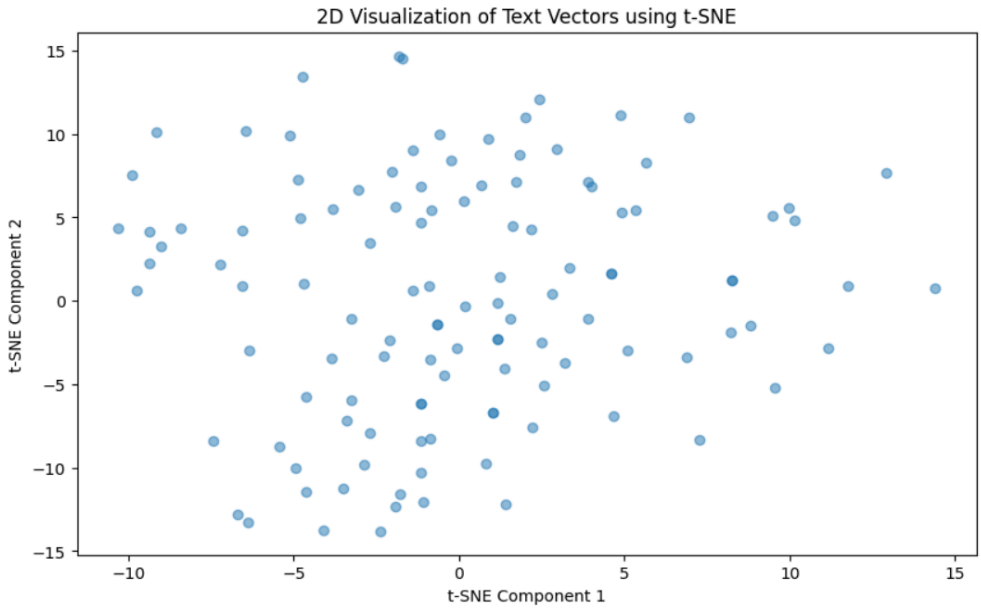
This system operates by first cleaning and vectorizing job descriptions using the pre-trained 'word2vec-google-news-300' model, which translates text into 300-dimensional vectors. These vectors are subsequently stored in a Pinecone vector database under the 'description' namespace, each associated with a job ID and title. Configured to use Euclidean

distance for similarity assessment, the database efficiently identifies how closely resumes match job descriptions. Upon querying with a cleaned resume, the system retrieves the top three nearest job descriptions based on vector similarity. It then presents the end user with the job titles and IDs of these highly matching job postings, enhancing the user's ability to find relevant job opportunities.

```
Titles with IDs:
ID: 48, Title: Data Scientist
ID: 42, Title: Data Scientist
ID: 87, Title: IT Support Service Analyst (M-F 8AM-4PM & M-F 9:30AM-5:30PM)
```

Output of the vector database model for a queried resume

To complement our understanding of the system's effectiveness, we utilized t-distributed Stochastic Neighbor Embedding (t-SNE), a dimension reduction technique, to visualize the vector data stored in the 'description' namespace of our vector database. By transforming the 300-dimensional vectors into a 2-dimensional format, this visualization allows us to observe the clustering of job postings directly



Section IV: Analysis of Results

The results reveal several key insights into the intricate relationship between resumes, job descriptions, and the AI-driven recruitment process. By delving into the linguistic characteristics of both resumes and job postings, patterns emerged that shed light on the strategies job seekers can employ to optimize their chances of being successful in the competitive job market. One notable finding is the disparity between the linguistic features highlighted by the different similarity models. The divergence between the GloVe embeddings model and the TF-IDF model underscores the complexity of AI-driven resume screening and suggests that different approaches may yield varying results. While the GloVe model emphasized data science roles, the TF-IDF model leaned towards information technology and systems positions. This nuanced understanding showed the importance of tailoring resumes to specific job roles and industries. We can clearly see that the top jobs from the job similarity model included a large amount of corporate information candidates would want transparency of before applying. Things like pay, and experience are proved to be necessary information to include in any job description looking to attract many applicants. The top resumes from the resume similarity model include more technical skills, which are obviously of interest to employers. In competitive fields like data science and information technology, stacking your resume with applicable technical skills is necessary to pass the AI screeners. Specifically, any skill mentioned in the description is imperative to have on your resume in order to pass the screeners.

Moreover, the classification models provide valuable insights into the predictive capabilities of AI systems in matching resumes to their ideal job categories. While the Random Forest Classifier demonstrated promising accuracy in identifying suitable job categories for resumes, the Support Vector Machine fell short, highlighting the challenges of employing certain

algorithms in high-dimensional feature spaces. The analysis of misclassifications offers opportunities for refinement, suggesting that a deeper understanding of the nuances in job descriptions and resume content is necessary for enhancing the accuracy of AI-driven recruitment systems. The two examples we showed in particular proved that sometimes an individual may not fully understand how their skills relate to different career opportunities. Having a more broad intended field will likely lead to more potential job matches, as many skills in these technical fields are transferable across disciplines. This predictive model can be further refined with a larger dataset. We only had around 200 resumes, therefore, there wasn't enough data for substantial training. A larger and more comprehensive dataset would allow these predictive models to learn better and improve the evaluation on unseen data. All in all, the findings underscore the evolving landscape of talent analytics and the need for continuous refinement and adaptation in leveraging AI technologies for recruitment purposes.

Observing the output of the vector database model, we find an intriguing scenario where a resume closely matches both a support role and a highly technical job. This prompts a critical examination of our current methodology, which utilizes all words from cleaned job descriptions to create embedding vectors. Such findings suggest the need for a more sophisticated approach to vector creation. To refine our system, we propose the development of a complex word extraction pipeline[3] that focuses on extracting key terms related to experience and skills, which hold significant weight in job descriptions. This pipeline will incorporate advanced NLP techniques including Named Entity Recognition, Part-of-Speech tagging, and Chunking. By doing so, we anticipate a substantial improvement in the precision of similarity searches between resumes and job descriptions, ensuring that candidates are matched with roles that truly correspond to their qualifications and expertise.

Section V: Next Steps

Moving forward, there are several avenues we can look into for future research and development. Firstly, further exploration into the nuances of AI-driven resume screening algorithms is necessary. For instance, fine-tuning these algorithms to account for industry specific language and job requirements could enhance their effectiveness in matching candidates with suitable roles. Additionally, ongoing refinement of classification models is essential to improve accuracy and address misclassifications, thereby increasing the reliability of AI-driven recruitment systems. Moreover, the integration of additional data sources and features, such as contextual information about job seekers and hiring organizations, could enrich the predictive capabilities of AI models. Collaborative efforts between researchers, industry stakeholders, and AI developers are crucial for advancing the state-of-the-art in talent analytics and ensuring that AI technologies continue to serve the evolving needs of the job market.

Sources:

1. https://arxiv.org/pdf/2307.03195
2. https://towardsdatascience.com/understanding-word-embeddings-with-tf-idf-and-glove-8acb63892032
3. https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html
4. https://github.com/RemeAjayi/ds-job-detective/blob/main/rule_based_skill_extraction.ipynb