Tech Job Satisfaction Prediction via Sentiment Analysis

Jason Bian: University of Texas as Austin

December 5, 2023

Abstract

Job markets are typically cyclic with boom and bust cycles. From a candidate's point of view, it is important to be able to predict the future outlook of different career paths as well as adjust existing career investments against market conditions. This report attempts to gauge the relative satisfactions ratings of different job families against job satisfaction. We will use data from OECD, Bureau of Labor Statistics, Linkedin, reddit, and glassdoor to create a joined dataset with the job satisfaction as the predictor. Various sentiment analysis packages are then used to build features and a final random forest model is used to predict glassdoor ratings per job family.

1 Research Background

We want to explore the relative job satisfaction of various candidates for different job families given differences in tenure, supply and demand for the position, and overall sentiment around the position and well as the company.

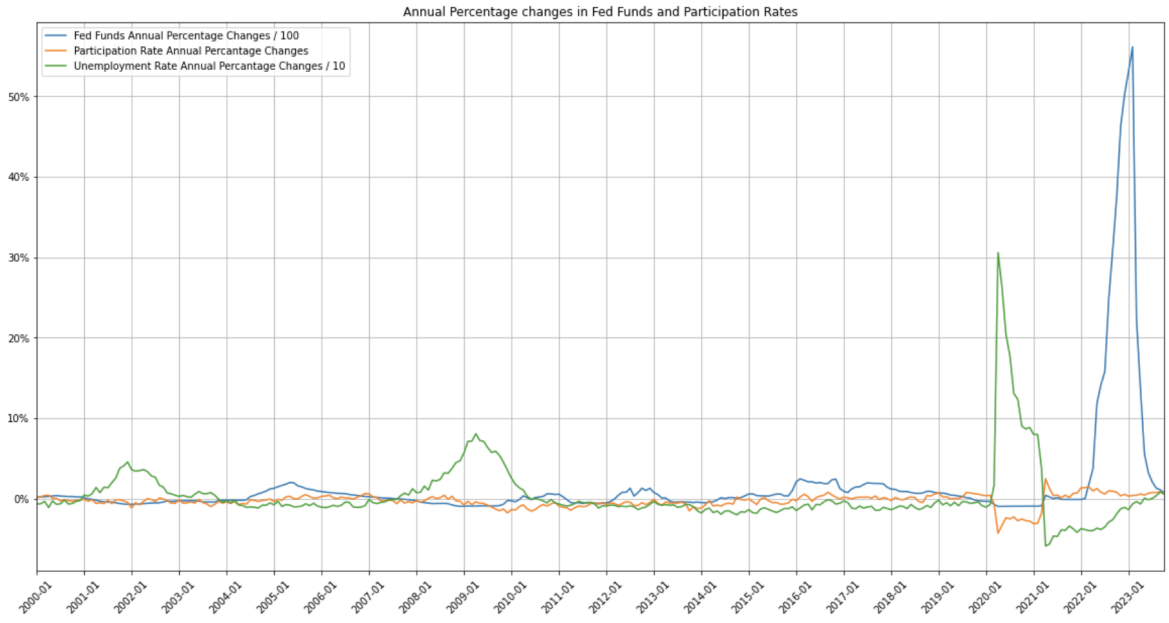
The relative stability and difficulty of job families given that career investments from candidates may be impacted by the increases in the application difficulties of certain positions over time. (Pandey, Dhrubalal 2019) found that millennials tend to job hop in job families with high employee transience, w However, candidates in positions with high transience face threats from other positions with high exit options or new entries into the job market. Additionally, (Steenackers, Kelly 2016) found that millennials and gen z have the largest tendency to job hop given generational attitudes.

On the financial side, the past year is an interesting period to analyze. The Beveridge curve tends to show an inverse relationship — unfilled vacancies tend to increase as unemployment rate decreases. The years where the vacancy rate is higher for the same unemployment rate constitute less efficient a labor market where 1) skills mismatches between what employers are looking for and what job seekers have, or 2) vacancies being filled predominantly by job switchers rather than the jobless, or 3) lower willingness on the part of the unemployed to take a given job.

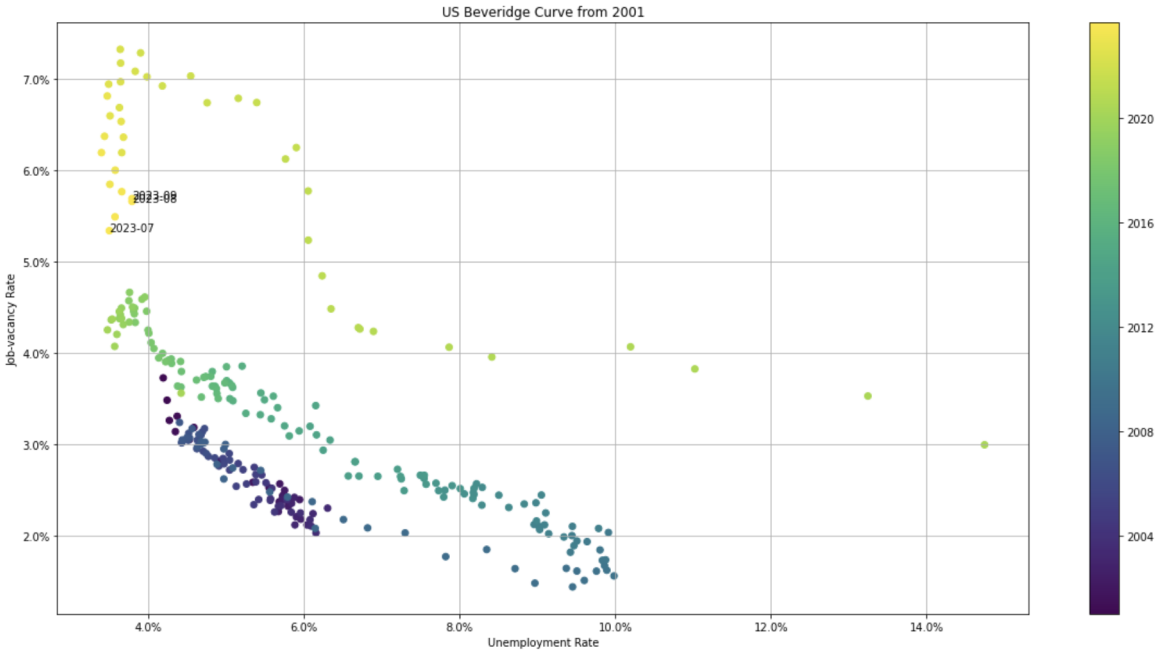
Given the recent tech recession and rising interest rates, we're in a unique market with high job vacancies and low inflation. This is especially pronounced in the tech sector, where critical roles remain unfilled, taken presumably by job switchers while entry roles remain with high barriers to entry. Additionally, there's also tech workers (Westlund, Steven 2008) who attempt to pivot due to new jobs to fulfill unmet expectations. Expectancy theory was able to model job hunters who had fallback options with intentions to quit and model key attributes that predicted attrition.

Hannon 2013 showed that pay was one of the main factors in reducing employee turnover in addition to other factors such as routinizing lower job stress. There was a significant negative relationship found between the satisfaction with the nature of work and turnover intentions.

Lastly, OECD data shows that we are in a period of recovering unemployment and job participation overall, yet the tech labor market is still facing supply and demand gap issues for entry workers (CNP16OV', 'CLF16OV', 'CE16OV' OECD Metrics). This report aims to explore that gap through the lens of job family sentiment.



2 Economic Background



The Beveridge Curve compares the unemployment rate to the job-vacancy rate (the number of job vacancies divided by the total labour force) and shows how this changes over time.

The Beveridge curve tends to show an inverse relationship — unfilled vacancies tend to increase as unemployment rate decreases. The logic being that when few are looking for a job, employers take longer to fill in their vacancies. Those years where the vacancy rate is higher for the same unemployment rate constitute less efficient a labor market where 1) skills mismatches between what employers are looking for and what job seekers have, or 2) vacancies being filled predominantly by job switchers rather than the jobless, or 3) lower willingness on the part of the unemployed to take a given job might be the reasons.

At the moment is is Reason 2 that's most likely the cause of today's "inefficiency" of the US job market where the same low unemployment rate as in 2014-2017 corresponds to a much higher vacancy rate. It is believed that about 80 percent of vacancies are filled by job switchers as of the end of 2022.

When vacancies are filled this way, the vacancy rate cannot start declining as when one job switcher fills a vacancy it automatically causes their past employer to open another.

Given that the barriers are filled this way, there's not a good way for entry level employees to access existing jobs as job hoppers typically aim for the same open positions as existing workers.

Additionally, the prevalence of job switching exacerbates the skills gap. Experienced workers moving between similar roles means there's less opportunity for upskilling or cross-skilling within organizations.

We aim to create natural clusters of popular tech career related job families (data engineer, data scientist, software engineer etc) and cross reference it with existing linkedin and indeed job datasets to identify clusters with low barriers of entry and clusters with high barriers of entry.

3 Problem Formulation

Given this, we are looking to measure job family satisfaction from the below set of global and local attributes at a job family level across different time-frames. We also include additional features from each of the datasets used to generate the below metrics.

1. **Vacancy Rate (Global Attribute):** (aka Unfilled Vacancies/Labor Force Ratio): The number of unfilled vacancies as a percentage of the labor force level.

$$\frac{|\text{Unfilled vacancies}|}{|\text{Labor Force Level}|}$$

2. **Job Sentiment (Global Attribute):** The average market sentiment over the current time period

3. **Unemployment Rate (Global Attribute):** Current BLS provided unemployment rates

4. **Job Difficulty (Job Family Attribute):** Measured as number of applicants/total postings.

$$\frac{|\text{Number of Applicants}|}{|\text{Total Postings}|}$$

5. **Job Sentiment (Job Family Attribute):** The job family specific sentiment over the current time period

4 Data

Job satisfaction was identified as the main reason for career changes. Given the current count of unfilled vacancies, we want to explore the relationship between different features against the various tech job families with job satisfaction as the response. We will perform the random forest at two intervals, one between 2022-10 to 2023-04 and one between 2023-04 and 2023-10.

4.1 Data:

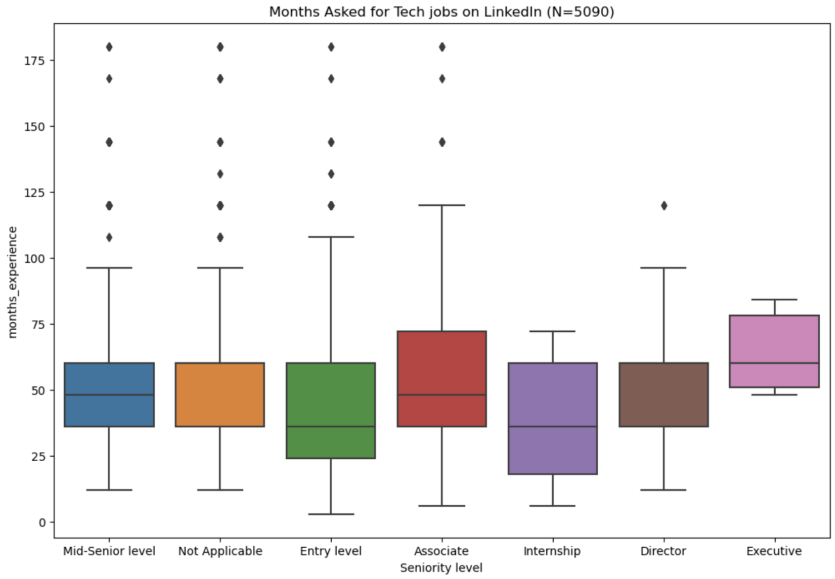
To investigate the application of ML in finding job family clusters, we will use web scrappers to pull 1 year of

1. Linkedin job postings

2. 8 different career Subreddits

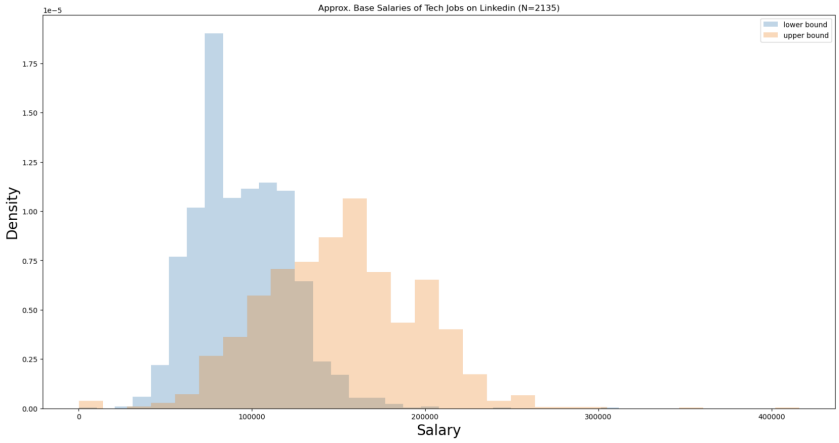
3. Glassdoor job reviews

and use nlp post-processing to assign each job to the closest job family entity. Linkedin data was used as the base dataset for job family entities, as the api provided ready to use job family IDs. Both the glassdoor and subreddit datasets were then joined against the company, job-title, and the location keys to create additional features.

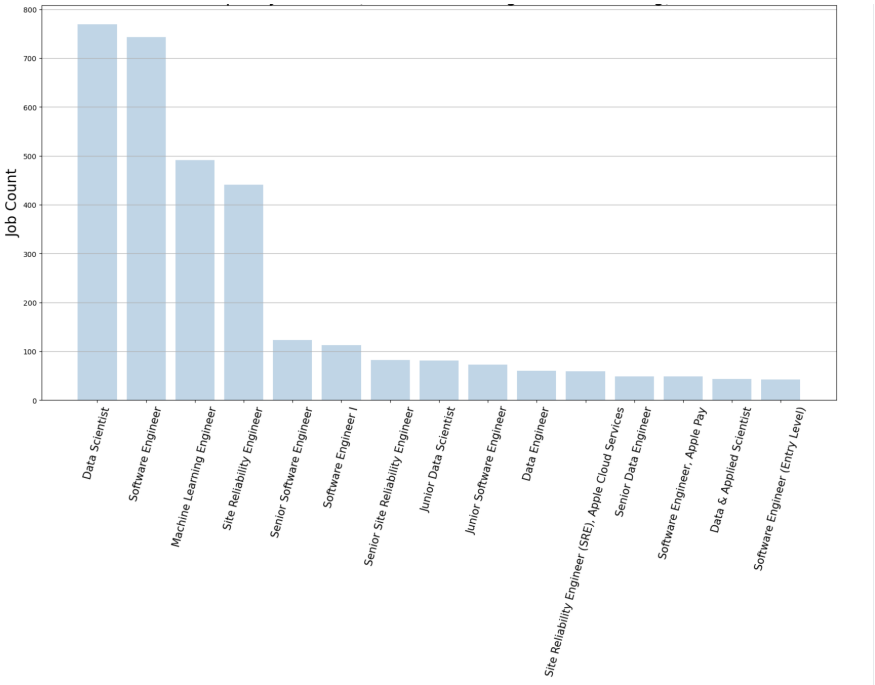


4.2 Linkedin Dataset:

Looking at the salaries of the scraped linkedin dataset:



Lastly, we plot the total counts for different job families within the linkedin dataset:



Given that we have data from multiple sources, we need to join them via different dimensions. The Linkedin job title key column was then manually assigned into 6 top entities using TF-IDF (See 4.3).

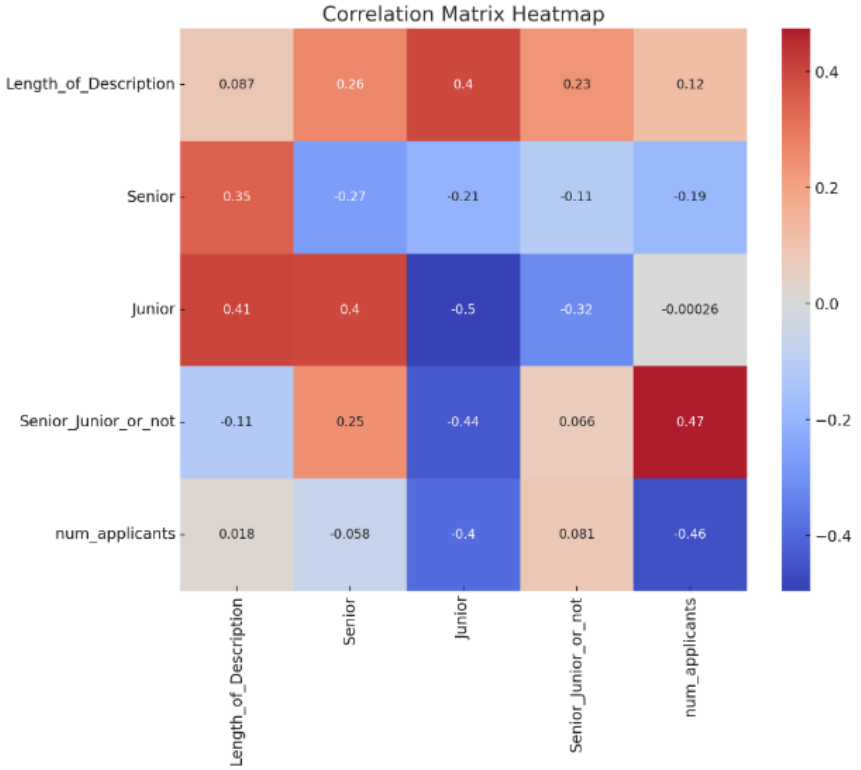
Column Name	Data Type
Job Title	string (nullable = false, primary-key = true, nlp-post-processing = true)
Salary Estimate	string (nullable = true)
Job Description	string (nullable = false, primary-key = true, nlp-post-processing = true)
Rating	string (nullable = true, response)
Company Name	string (nullable = true)
Location	string (nullable = true)
Headquarters	string (nullable = true)
Size	string (nullable = true)
Founded	string (nullable = true)
Type of ownership	string (nullable = true)
Industry	string (nullable = true)
Sector	string (nullable = true)
Revenue	string (nullable = true)
Competitors	string (nullable = true)
date_data_created	string (nullable = true)

4.3 Glassdoor Dataset:

Next, we join the dataset with the glassdoor data to gain additional features. The job title was then normalized to the below categories after pulling from the glassdoor api: Glassdoor api. Senior and Junior entities were extracted using the ntlk python package from the job descriptions.

Column Name	Data Type
Job_Title	string (nullable = false, primary-key = true, nlp-post-processing = true)
Company	string (nullable = false, primary-key = true)
Location	string (nullable = true)
Number_of_Applicants	string (nullable = true)
Description	string (nullable = false, nlp-post-processing = true)
Length_of_Description	string (nullable = true)
Senior	string (nullable = true)
Junior	string (nullable = true)
State	string (nullable = true)
date_data_created	string (nullable = true)

To see the correlations between the above features and investigate potential segmentation, we can create a correlation matrix. We see that senior positions have much more applicants compared to junior positions. Additionally, junior positions have much longer descriptions compared to senior positions. There's also a strong correlation between number of applicants and positions without senior or junior entity mentions (Senior-Junior-or-not)



4.4 Reddit Dataset:

Finally, we scrape various subreddits and join a "sentiment feature for each subreddit"). The below scraper parameters were used for the two designated 6 month time periods using the Praw api:

- **upvoteRatio** = 0.70 % upvote ratio for post to be considered, 0.70
- **ups** = 20 # of upvotes, post is considered if upvotes exceed this number
- **limit** = 500 # define the limit per subreddit, comments 'replace more' limit
- **upvotes** = 2 # of upvotes, comment is considered if upvotes exceed this number

Attribute	Description
author	Provides an instance of Redditor.
author-flair-text	The text content of the author's flair, or None if not flaired.
clicked	Whether or not the submission has been clicked by the client.
comments	Provides an instance of CommentForest.
created-utc	Time the submission was created, represented in Unix Time.
score	The number of upvotes for the submission.
selftext	The submissions' selftext - an empty string if a link post.
spoiler	Whether or not the submission has been marked as a spoiler.
stickied	Whether or not the submission is stickied.
subreddit	Provides an instance of Subreddit.
title	The title of the submission.
upvote-ratio	The percentage of upvotes from all votes on the submission.

4.5 Methods: Sentiment Analysis:

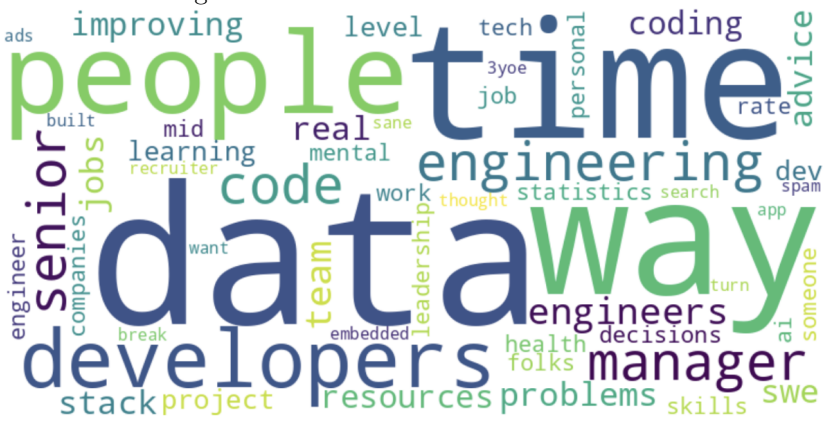
(Shamrat, Mehedi 2020) uses k-means to iterate through different job entities and detect pockets with demand supply gaps.(Lukauskas, Mantas 2023) scraped the largest portals from Lithuanian job advertisements and searched for specific words and phrases. By using named entity recognition and term frequency-inverse document frequency (TF-IDF):

$$w_{i,j} = tf_{i,j} \times \log\left(\frac{N}{df_i}\right)$$

We find key entities in the job title and within descriptions to normalize job families between the datasets. Other methods include bag of words and doc2vec, where text vectorization is used as alternatives to sentence transformers. Using TF-IDF via the nltk package, we identified the below 6 set of top entities:

- Data scientist
- Junior Software Engineer
- Data Engineer
- Experienced Software Engineer
- Site Reliability Engineer
- Machine Learning Engineer

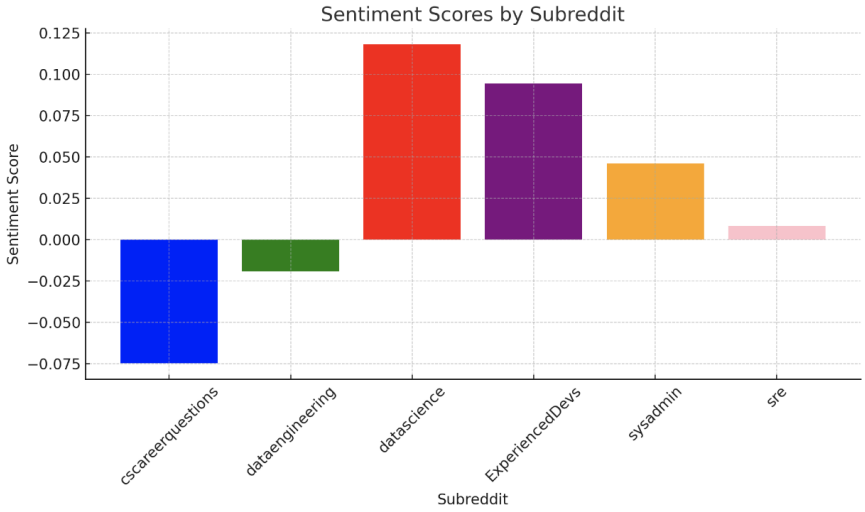
Next we calculate the sentiment of each subreddit and standardize it relative to the sentiment of existing glassdoor review with the below algorithm. Nltk was used to tokenize reddit comments with top words from the bag of works tokenization as shown below.



Sentiment scores was then calculated per subreddit for each of the two time frames and joined to the main dataset.

Algorithm 1 Calculate Subreddit Sentiment

$subreddit \leftarrow reddit.subreddit(sub)$
$top\_python \leftarrow subreddit.top(filter = time_interval)$
for all $submission \in top\_python$ **do**
  $flair \leftarrow submission.link\_flair\_text$
  $author \leftarrow submission.author.name$
  if $(submission.upvote\_ratio \geq upvoteRatio)$ **and** $(submission.ups > ups)$ **and** $((flair \in post\_flairs)$ **or** $(flair$ is $None))$ **and** $(author \notin ignoreAuthP)$ **then**
    $sentiment\_score \leftarrow sia.polarity\_scores(submission.title)['compound']$
    $subreddit\_sentiments[sub]['total\_sentiment'] \mathrel{+}= sentiment\_score$
    $subreddit\_sentiments[sub]['count'] \mathrel{+}= 1$
  end if
end for
$average\_sentiments \leftarrow \{\}$
for all $(sub, data) \in subreddit\_sentiments.items()$ **do**
  if $data['count'] > 0$ **then**
    $average\_sentiments[sub] \leftarrow \frac{data['total\_sentiment']}{data['count']}$
  else
    $average\_sentiments[sub] \leftarrow 0$
  end if
end for=0



The mapping of subreddits to entites is as follows:

- Data scientist - r/datascience
- Junior Software Engineer - r/cscareerquestions
- Data Engineer - r/dataengineering
- Experienced Software Engineer - r/ExperiencedDevs
- Site Reliability Engineer - r/sre
- Machine Learning Engineer - r/MachineLearning

The sentiment scores per time period were then joined into the main dataset via the job family and time keys primary keys. Additionally, the job descriptions for both linkedin posts and glassdoor posts were tokenized and the relevant sentiment calculated at the primary key level.

4.6 Validation:

Evaluation of other nlp packages on the 1 year dataset was performed using 5-fold validation on the below packages:

1. nltk
2. spacy
3. textblob
4. vader

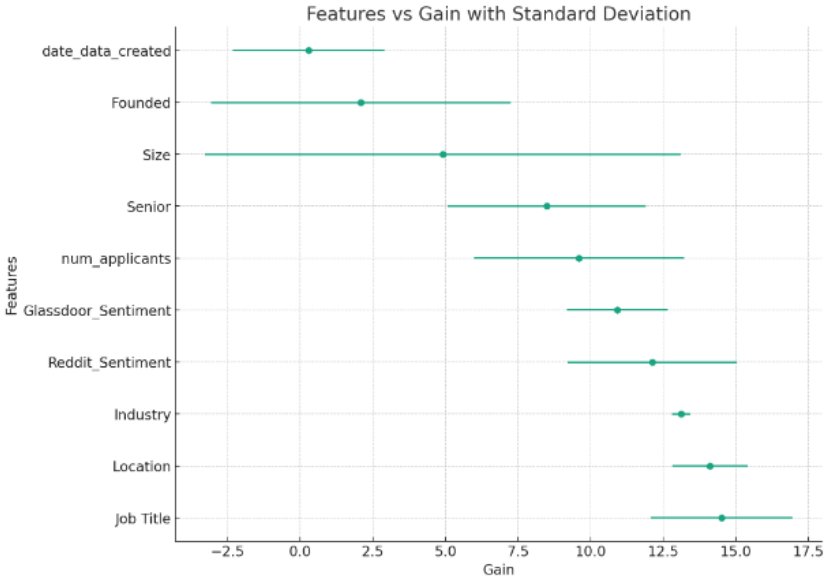
   Table 1 shows the performance of linear regression and gradient boosting in a 5-fold train and test split with the Glassdoor job satisfaction as the response.

Algorithm	**Sentiment Engine**	**Train MAE**	**Test MAE**	**Train R$\hat{2}$**	**Test R$\hat{2}$**
Linear Regression	nltk	0.1436	0.2479	0.2147	0.0924
Gradient Boosting	nltk	0.3342	0.3272	0.0885	0.0112
Linear Regression	spacy	0.3375	0.3522	0.1143	0.0124
Gradient Boosting	spacy	0.4292	0.4207	0.0934	0.0118
Linear Regression	textblob	0.5451	0.5539	0.1149	0.0124
Gradient Boosting	textblob	0.3471	0.3495	0.0896	0.0120
Linear Regression	vader	0.4501	0.4394	0.1151	0.0126
Gradient Boosting	vader	0.3296	0.3535	0.0873	0.0110

Table 1: Model Performance with Scaled and Randomized Values

Given that nltk had the best performance, we then run the ntlk model for the two time periods to see SHAP contributions to see the importance value of the total features.

2023-10-01 to 2023-04-01 SHAP Scores:



2023-04-01 to 2022-10-01 SHAP Scores:

4.7 Results and Conclusion:

The second half of the 2023 was characterized by decreasing unemployment (See Research background) and increasing job vacancy rates. The higher contribution of the num-applicants field to job satisfaction could indicate reluctance to job hop during the improving recession. Additionally, Job title has consistently been a contributing predictor to job satisfaction. This is supported by the reddit sentiment categories, where the r/datascience and r/ExperiencedDevs subreddits typically have higher satisfaction as employees reach a stable point in their career.

Meanwhile, cscareerquestions is an early career subreddit and many job applicants are actively on reddit looking for employment. This is shown by the junior or senior categorical classification features in the SHAP plot. r/ExperiencedDevs also may use the subreddit for advice and career development topics rather than for job hunting topics. Lastly, location and company name contribute to the Glassdoor satisfaction as some companies are generally better to work for then others.

4.8 References:

Lundberg, S., Lee, S.-I. (2017). A unified approach to interpreting model predictions. arXiv . Retrieved from https :// arxiv .org / abs / 1705 .07874 DOI: 10.48550/ARXIV.1705.07874

Phu, N., Le, H. (2021). Determinants of job hopping behavior: The case of information technology sector. International Journal of Law and Management. https://doi.org/10.2139/ssrn.3818661

Bureau of Labor Statistics. (2023). Job openings and unemployment: Beveridge Curve. Retrieved from https://www.bls.gov/charts/job-openings-and-labor-turnover/job-openings-unemployment-beveridge-curve.htm

Shamrat, F. M., Tasnim, Z., Mahmud, I., Jahan, N., Nobel, N. (2020). Application of K-means clustering algorithm to determine the density of demand of different kinds of jobs. [Journal Name], 9, 2550-2557.

Lukauskas, M., Šarkauskaitė, V., Pilinkienė, V., Stundziene, A., Grybauskas, A., Bruneckienė, J. (2023). Enhancing skills demand understanding through job ad segmentation using NLP and clustering techniques. Applied Sciences, 13. https://doi.org/10.3390/app13106119

Glassdoor. (2023). Glassdoor API. Retrieved from https://www.glassdoor.com/developer/index.html

Praw. (2023). Praw API. Retrieved from https://praw.readthedocs.io/en/stable/

Alam, A., Asim, D. M. (2019). Relationship between job satisfaction and turnover intention. International Journal of Human Resource Studies, 9, 163. https://doi.org/10.5296/ijhrs.v9i2.14618

Pandey, D. (2019). Job hopping tendency in millennials. NCC Journal, 4(1), 41-46. https://doi.org/10.3126/nccj.v4i1.2

Steenackers, K., Guerry, M.-A. (2016). Determinants of job-hopping: An empirical study in Belgium. International Journal of Manpower, 37, 494-510. https://doi.org/10.1108/IJM-09-2014-0184

Westlund, S. (2008). Retaining talent: Assessing job satisfaction facets most significantly related to software developer turnover intentions. International Journal of Information Technology and Management - IJITM, 19.

Lundberg, S., Lee, S.-I. (2017). A unified approach to interpreting model predictions. arXiv. https://doi.org/10.48550/ARXIV.1705.07874

The pandas development team. (2020, February). pandas-dev/pandas: Pandas (Latest version). Zenodo. https://doi.org/10.5281/zenodo.3509134

Honnibal, M., Montani, I. (2017). spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing.

Hutto, C. J. Gilbert, E. (2014). VADER: A Parsimonious Rule-Based Model for Sentiment Analysis of Social Media Text. A rule-based model for sentiment analysis of social media text, known as VADER (Valence Aware Dictionary and sEntiment Reasoner)..

Loria, S. (2018). textblob Documentation. Release 0.15, 2.