Data Job Analysis & Modeling

Anuja Panthari & Sai Velicheti
INLS 642 Fall 2024

Project Motivation

01 Projected Job Growth

Demand for data scientists is skyrocketing, with a projected 35% jump in job openings between 2022 and 2032

02 Current Market Frenzy

Job openings are growing at a sluggish pace (7%) compared to the recent surge in applications (31%)

03 Remote Work Reshaping Hiring Trends

The pandemic-era shift to remote work has solidified as a norm, with over 60% of data science roles now offering flexible or fully remote options.

Project Outline

01 Data Scraping Process

02 Brief Data Description

03 Data Modeling

04 Key Takeaways

Dataset Descriptions

01

DATA SOURCE

JobFunnel

02

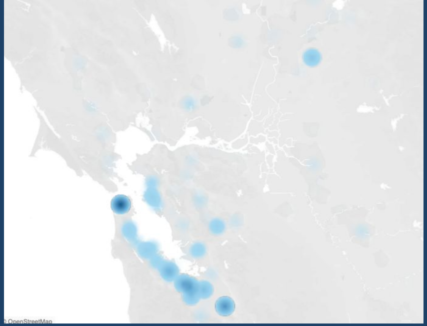
KEY FEATURES

- Job Title
- Company
- Location
- Blurb
- Wage
- Remoteness
- State

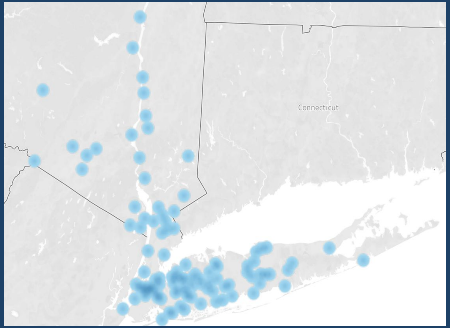
High Level Analytics

Total Jobs: 10,604

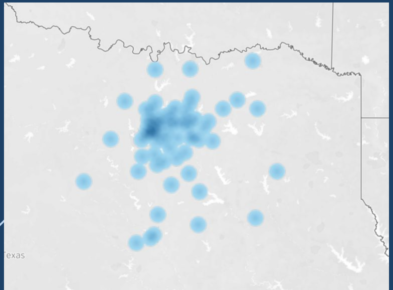
San Francisco: 3,081



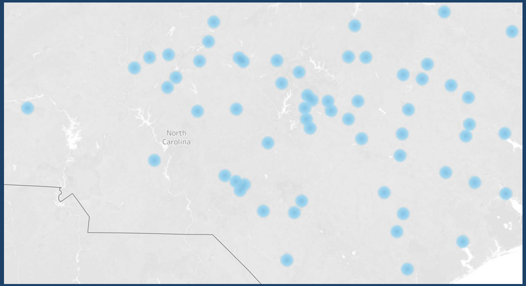
New York City: 2,973

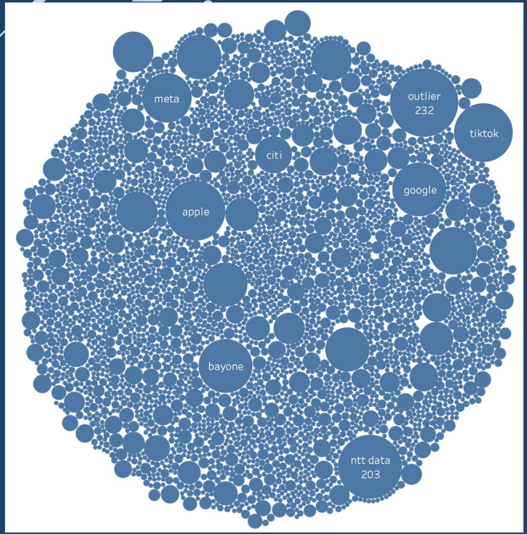


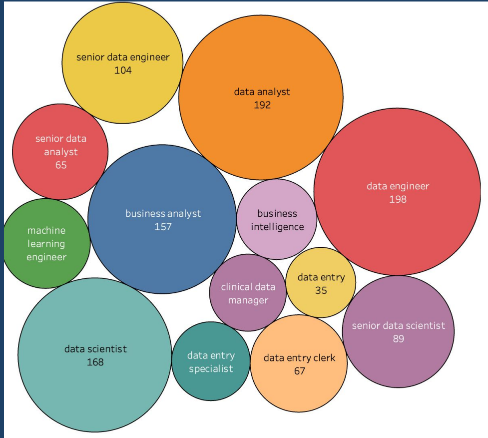
Dallas: 2,710



Raleigh: 1,796







Query:

- “Data” keyword
- 4 major city locations
- 80 km radius
- Last 1 month (11/2 - 11/27)

Analysis Goals

- **Main Goal:** Data Segmentation - Are there patterns in the data we can glean?

Method 1: Supervised learning

Artificially creating job level labels using chatGPT and cross validating using “gold standard”

Method 2: Unsupervised learning

Utilizing PCA and k-means clustering

Method 1: Supervised Learning

- We are interested in acquiring job level feature in dataset -- but we don't have it!
- Can we predict job level using the dataset metrics?
- ChatGPT used **keywords** to segment jobs in 3 levels
- Retrieval Augmented Generation (RAG model) to check efficacy of ChatGPT
- Manually Labeled 600 data points for quality control
- How close is the chatGPT prediction of job level to our "gold standard"

Method 1: Supervised Learning

- We trained out model on a subset of our data that was “labeled” by chatGPT
- Using logistic regression, we trained a model using vectorized **title** & **blurb** columns
- We found pretty low accuracy when checked on our manually labeled test set
- What does this mean?
  - ChatGPT’s way of labeling does not align with our method of labeling
  - Why? Metrics used to define seniority differed between training & test sets

```
▾ LogisticRegression
LogisticRegression(multi_class='multinomial')
```

```
logistic regression, accuracy on test set: 0.3422370617696 1605
```

Defining Job Level

A brief philosophical caveat

- Leads to a larger discussion about how to define job level? By years of experience? Company? Title? Salary? A combination?

- **The problem**: no clear way of defining job level that all companies abide by

Examples: *How would you segment these jobs?*

"Senior Software Engineer" requires 6 years of experience

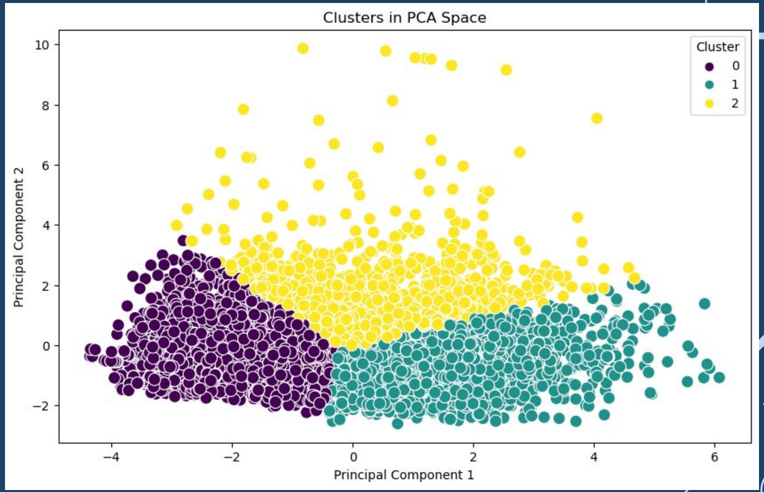
"Database Manager" has salary of $75,000, requires 15+ years of experience

Amazon "Data Scientist" in CA that has a $250,000 salary requiring 2-3 years of experience

- Since "years of experience" was how we decided to manually label out dataset, and that metric was not in our dataset, the training and test sets would never align!
- Unless years of experience mapped to keywords, which we now know it doesn't.

Method 2: Unsupervised Learning

- Are there any clear patterns found in the data?
- Vectorized, and selected features using chi-square values within the title, blurb & company
- Encoded state into dummy variables
- Used PCA to reduce sparse matrix
- K-Means Clustering Methods: (1) Job title, Blurb, Company & (2) Job title, Blurb, Company, State Salary



Key Takeaways

- Defining Job Level is hard task -- variable standardization is needed to build a more effective model

- There was an interesting pattern that was found in PCA values when using title, blurb and company but NOT when adding in state and salary→ What does this mean?

- Further look into pattern in PCA plot