# CS5530- Principles of Data science

# University of Missouri-Kansas City

# Analysis of Big Data Job postings

# Harika Thippireddy(16328121)

# Sathvik Reddy Anumasu(16336266)

# Gopi Venkata Sudheer Arikatla(16336237)

# Durga Akhila Bhuvaneshwari Eripina(16342213)

## Introduction

The need for qualified candidates in Big Data jobs is always growing in today's hectic corporate climate. The Human Resources (HR) division is essential to a company's ability to draw in and keep the best individuals for its Big Data requirements. Acknowledging the difficulties in filling these specialized roles, a large-scale Big Data initiative is suggested to examine a large quantity of data taken from actual job postings that are posted online.

## Problem Statement and Motivation

Recruiting skilled Big Data professionals poses a significant challenge for HR departments due to the dynamic nature of the field and the lack of effective methods for understanding the nuanced requirements of various job roles. Traditional recruitment approaches often lead to suboptimal hiring decisions, prolonged time-to-fill positions, and mismatches between candidates and job expectations. The absence of a systematic analysis of real-world job posts exacerbates the difficulty in identifying trends and patterns in the Big Data job market. The motivation for this Big Data project lies in the critical need to acquire and retain skilled professionals for driving innovation and maintaining competitiveness in the rapidly evolving business landscape. Recognizing the limitations of traditional recruitment practices, the project aims to leverage Big Data analytics to empower HR departments with data-driven insights. The goal is to revolutionize the recruitment process, enabling HR professionals to make informed decisions and strategically build a workforce proficient in Big Data skills, aligned with the diverse requirements of job families within the Big Data domain.

## Research Objectives

In the dynamic landscape of Big Data, where the demand for skilled professionals is ever-expanding, the challenges faced by HR departments in effective recruitment necessitate a targeted and data-driven approach. This research embarks on a comprehensive investigation with the following key objectives, aiming to not only decipher the intricate nuances of the Big Data job market but also equip HR professionals with actionable insights to enhance their recruitment strategies and decision-making processes.

The following are objectives of the project:

* Identify and categorize distinct job families within the Big Data domain based on responsibilities and emerging roles.
* Employ data analytics to group similar Big Data skills from job posts, revealing patterns and trends.
* Develop a competency framework to characterize each job family based on the proficiency level required for specific skills.
* Analyze job posts to understand the relative importance and demand for different Big Data skills.
* Study temporal patterns to anticipate evolving trends in Big Data skills demand.
* Formulate data-driven recommendations for targeted recruitment strategies based on job families and skills clusters.
* Provide insights into potential skill development areas, aiding in the design of training programs aligned with market needs.
* Ensure the relevance and accuracy of findings by validating insights with industry experts.
* Create an interactive tool for HR professionals to explore and apply project insights in making informed recruitment decisions.

## Project Scope and Direction

The scope of this Big Data analytics project is strategically designed to address the challenges faced by HR departments in recruiting for Big Data roles. The project will encompass the following key elements:

* Identifying and categorizing distinct job families within the Big Data domain based on emerging roles and responsibilities.
* Utilizing advanced data analytics to cluster and understand patterns in Big Data skills, providing insights into skills highly valued by companies.
* Developing a competency framework to characterize each job family based on the proficiency levels required for specific skills.
* Analyzing job posts to assess the relative importance and demand for different Big Data skills, offering insights into evolving industry needs.
* Visualizing the Big Data skills landscape through a heatmap to highlight clusters and their prevalence across job families.
* Exploring temporal patterns to anticipate evolving trends in Big Data skills demand, enabling proactive recruitment strategies.
* Formulating data-driven recommendations for targeted recruitment strategies based on identified job families and skills clusters.
* Providing insights into potential skill development areas, allowing organizations to design targeted training programs.
* Ensuring credibility and relevance through validation with industry experts, enhancing the practical applicability of research outcomes.
* Creating an interactive tool to empower HR professionals with actionable insights for effective Big Data recruitment.

## Data Collection and Preprocessing

## Data Set

## 

This dataset comprises information extracted from 1,500 job postings related to data science from Glassdoor.com. The data encompasses essential details about each job listing, facilitating comprehensive analysis and insights into the data science job market.

Columns

Job Title: The title of the data science job position.

Salary Estimate: The estimated salary range associated with the job.

Job Description: A detailed description of the responsibilities and requirements for the job.

Rating: The company's rating on Glassdoor.

Company Name: The name of the hiring company.

Location: The geographical location of the job.

Size: The size of the company in terms of employees.

Founded: The year the company was founded.

Type of Ownership: The ownership structure of the company (e.g., public, private).

Industry: The industry in which the company operates.

Sector: The sector to which the company belongs.

Revenue: The revenue of the company

1. **Data Cleaning**
2. **Drop Duplicated and NaN Values:**

Data cleaning is a crucial step in the data preparation process, ensuring that the dataset is free from inconsistencies and missing values that could potentially impact the accuracy of analyses and models. Two fundamental operations in data cleaning are dropping duplicates and filling NaN (Not a Number) values.

**Dropping Duplicates:**

Duplicate entries in a dataset can skew analyses and provide redundant information. The process of dropping duplicates involves removing identical rows, leaving only unique instances. This is particularly important when dealing with datasets that might have been merged or aggregated, resulting in repeated observations.

**Filling NaN Values:**

Missing or NaN values in a dataset can disrupt analyses, as many statistical and machine learning algorithms struggle to handle missing data. Filling NaN values is the practice of replacing these missing values with specific data, such as a constant, the mean, median, or values from adjacent observations.

A screenshot of a computer

Description automatically generated

1. **Parsed numeric data out of salary column as min, max and avg \_salary:**

Extracting meaningful information from salary data is a crucial step in preparing a dataset for analysis. In this context, parsing numeric data from a salary column and creating separate columns for minimum (min\_salary), maximum (max\_salary), and average (avg\_salary) salaries is a common practice. Additionally, when dealing with salaries specified on an hourly basis, it's essential to consider this aspect in the data processing. Furthermore, creating new columns to differentiate between employer-provided salaries and hourly wages provides additional insights into compensation structures.

**Parsing Numeric Data and Calculating Salaries:**

Parsing numeric data from a salary column involves extracting numerical values, considering different formats and units such as annual salaries and hourly wages. Creating separate columns for minimum, maximum, and average salaries enables a more granular analysis of compensation ranges.

**Adding Columns for Employer-Provided Salary and Hourly Wages:**

Distinguishing between employer-provided salaries and hourly wages allows for a more nuanced analysis of compensation structures within the dataset.

A screenshot of a computer

Description automatically generated

1. **Parsed rating out of company text and removed undesired characters:**

Parsing and cleaning rating data from a company text column is essential for obtaining structured and numeric information for analysis. This process involves extracting numerical values from the text and removing any undesired characters, ensuring the resulting data is suitable for quantitative analysis.

A screenshot of a computer

Description automatically generated

In this code:

df.apply(lambda x: x['Company Name'] if x['Rating'] < 0 else x['Company Name'][:-3], axis=1): Adjusts 'company\_txt' based on the condition you specified.

df['company\_txt'].str.rstrip('\n'): Removes trailing newline characters ("/n") from the 'company\_txt' column.

1. **Made a new column for company state and cleaned it.**

# We have created a new column 'job\_state' based on the 'Location' column in your DataFrame. This process typically involves extracting state information from the 'Location' column and then performing cleaning steps to standardize the representation of states.

# A screenshot of a computer Description automatically generated

# Transformed founded date into age of company

# Transforming the founded date into the age of the company is a useful step in data analysis, providing a more relevant and standardized metric for understanding the companies' timelines.

# 

# Made columns for if different skills were listed in the job description

# Creating binary columns to indicate the presence or absence of specific skills in the job description is a common and valuable preprocessing step for data analysis or machine learning tasks.

# A screenshot of a computer Description automatically generated

# After running this code, we will have additional binary columns in your DataFrame indicating the presence of specific skills. These columns can be used for further analysis, visualization, or as features in machine learning models to understand the importance of certain skills in job descriptions.

# Column for simplified Job Title

# Creating a simplified job title column can be beneficial for grouping and analyzing similar roles in a more concise manner.

# A screenshot of a computer program Description automatically generated

# Column for Seniority

# Creating a column for seniority based on both job title and job description can provide additional insights into the level of positions.

# A screenshot of a computer program Description automatically generated

# A screenshot of a computer Description automatically generated

# A screenshot of a computer Description automatically generated

# A screenshot of a computer program Description automatically generated

# Column for Job Description length

# If we want to display only the 'Job Description' column along with the newly created columns for description length in terms of letters and words, we can modify the print statement accordingly.

# A screenshot of a computer Description automatically generated

# Checking the others columns

# A screenshot of a computer program Description automatically generated

# Finally Exporting

# We are saving our cleaned DataFrame, df\_out, to a CSV file named 'data\_cleaned.csv'. This is a good practice for preserving your processed data for further analysis or sharing.

# A white rectangular object with blue dots Description automatically generated

## Conceptual Diagram

## A screenshot of a computer screen Description automatically generated

## Exploratory Data Analysis (EDA)

## Identify various Big Data job families in the given dataset

## To identify various Big Data job families in the given dataset description, you can perform text analysis or natural language processing (NLP) techniques. tokenize\_and\_filter is a function that tokenizes a job title, converts tokens to lowercase, removes stop words, and returns the filtered tokens. Apply the tokenize\_and\_filter function to the 'Job Title' column and create a new column 'Tokenized Title' containing the filtered tokens.

## Identify common terms which create a series of common terms from the tokenized job titles. Define a threshold for common terms to set a threshold to determine which terms are considered common and indicative of a job family.

## A screenshot of a computer program Description automatically generated

## This code assumes that the 'Job Title' column contains job titles, and it identifies common terms indicative of Big Data job families. The identified families and their associated job titles are printed for further analysis.

## A white screen with black text Description automatically generated

## A screenshot of a computer Description automatically generated

## Identify homogeneous groups of Big Data skills that are highly valued by companies

## Identifying homogeneous groups of Big Data skills highly valued by companies typically involves analyzing the skills mentioned in job descriptions. The code you've posted earlier tokenizes job titles, but for skill analysis, we need to focus on the 'Job Description' column.

## 

## The code utilizes the NLTK library to analyze job descriptions and identify common Big Data skills highly valued by companies. It tokenizes and filters words in the 'Job Description' column, creating a new column 'Tokenized Description.' Common skills are identified based on frequency, and a threshold is set to determine highly valued skills. A new column 'Big Data Skills' is created, listing relevant skills for each job description. The code concludes by printing unique Big Data skills mentioned across the dataset, providing insights into valued skills in the context of Big Data job roles. Adjustments to the threshold and tokenization can be made based on specific dataset characteristics.

## A screenshot of a computer code Description automatically generated

## Characterize each Big Data job family according to the level of competence required for each Big Data skill set

## A screenshot of a computer code Description automatically generated

## The code tokenizes and filters job descriptions to identify common Big Data skills. For each job description, it assesses the level of competence required for each identified skill. This characterization is based on the frequency of skill mentions within the job description, providing insights into the importance of specific skills for different job families. The code then creates a new column 'Big Data Skills' to capture the identified skills and their corresponding levels of importance. The result is a nuanced understanding of skill requirements across various Big Data job families, aiding in the analysis of competency levels associated with each skill set. Adjustments can be made to the code to further refine skill characterization based on specific dataset nuances.

## A graph of different colored squares Description automatically generated

## Label Encoding:

## Label encoding is commonly used when dealing with categorical variables in machine learning models. It helps in transforming categorical data into a format that can be fed into machine learning algorithms.

## 

## Define features and target variable:

## 

## These features and the target variable will be used to train a machine learning model. Make sure that your dataset is appropriately split into training and testing sets before training the model.

## Perform one-hot encoding on the 'location' feature:

## Perform one-hot encoding on the 'location' feature, you can use the get\_dummies function from pandas.Interpret the results and consider any necessary adjustments or improvements.

## 

## Train Test Split:

## To split your dataset into training and testing sets for machine learning, you can use the train\_test\_split function from the sklearn.model\_selection module. This function randomly shuffles and splits your data into two sets: one for training your model and one for evaluating its performance.

## 

## References:

## Hariri, R.H., Fredericks, E.M. and Bowers, K.M., 2019. Uncertainty in big data analytics: survey, opportunities, and challenges. Journal of Big Data, 6(1), pp.1-16.

## Sestino, A., Prete, M.I., Piper, L. and Guido, G., 2020. Internet of Things and Big Data as enablers for business digitalization strategies. Technovation, 98, p.102173.

## Brown, P. and Souto-Otero, M., 2020. The end of the credential society? An analysis of the relationship between education and the labour market using big data. Journal of Education Policy, 35(1), pp.95-118.

## Ikotun, A.M., Ezugwu, A.E., Abualigah, L., Abuhaija, B. and Heming, J., 2023. K-means clustering algorithms: A comprehensive review, variants analysis, and advances in the era of big data. Information Sciences, 622, pp.178-210.

## Reddy, G.T., Reddy, M.P.K., Lakshmanna, K., Kaluri, R., Rajput, D.S., Srivastava, G. and Baker, T., 2020. Analysis of dimensionality reduction techniques on big data. Ieee Access, 8, pp.54776-54788.

## Deepa, N., Pham, Q.V., Nguyen, D.C., Bhattacharya, S., Prabadevi, B., Gadekallu, T.R., Maddikunta, P.K.R., Fang, F. and Pathirana, P.N., 2022. A survey on blockchain for big data: Approaches, opportunities, and future directions. Future Generation Computer Systems, 131, pp.209-226.

## Dai, H.N., Wang, H., Xu, G., Wan, J. and Imran, M., 2020. Big data analytics for manufacturing internet of things: opportunities, challenges and enabling technologies. Enterprise Information Systems, 14(9-10), pp.1279-1303.