databricks

Employee_rating

(https://databricks.com)

DESCRIPTION

Objective: To use Hive features for data analysis and sharing the actionable insights into the HR team for taking corrective actions.

Problem Statement: The HR team is surfing social media to gather current and ex-employee feedback or sentiments. This information gathered will be used to derive actionable insights and take corrective actions to improve the employer-employee relationship. The data is web-scraped from Glassdoor and contains detailed reviews of 67K employees from Google, Amazon, Facebook, Apple, Microsoft, and Netflix.

Domain: Human Resource

Analysis to be done: Exploratory analysis, to determine features and relationships impacting employee satisfaction and derive actionable insights by learning from the historical data

Content: This data set contains employee_review_data.csv separated into the following categories:

1. Index: Index number.
2. Company: Name of the company being reviewed.
3. Location: This dataset is global, as it includes the country& 39;s name in parenthesis [for example, "Toronto, ON(Canada)"]. However, if the location is the USA then it will only include the city and state [i.e. "Los Angeles, CA"].
4. Date Posted: Date posted will be in the following format MM DD, YYYY.
5. Job-Title: This string will also include whether the reviewer is a "Current" or "Former" Employee at the time of review. Both are fixed-length strings (“Current Employee ”and “Former Employee ”) followed by the role of reviewer.
6. Summary: Short summary of employee review.
7. Pros: Pros
8. Cons: Cons
9. Overall Rating: 1-5
10. Work/Life Balance Rating: 1-5
11. Culture and Values Rating: 1-5
12. Career Opportunities Rating: 1-5
13. Comp and Benefits Rating: 1-5
14. Senior Management Rating: 1-5
15. Helpful Review Count: A count of how many people found the review to be helpful/

Steps to perform:

Create a hive table partitioned by country and bucketed by year and also load the review.csv file. Note: Ensure that the right hive environment variable is set for bucket insert.

Impute the missing value (none) for all rating columns with a numerical value between 0 and 5. Note: For imputation, calculate the median for each of the 5 rating fields and create a new table.

- Work-balance stars
- Culture values stars
- Career opportunities-stars
- Comp-benefit-stars
- Senior-management-stars

Write the final relation schema to review.csv file in your HDFS home directory.Using the over-all rating fields display trend:

Globally by company Identify trends at 25%, 50%, 75%

- 2. Globally by company per year Identify trends at 25%, 50%, 75%
- 3. By company by country (Identify trends for each company by country Identify trends at 25%, 50%, 75%

Display the impact of employee status on rating a company using the overall-ratings field by the company by year.

Display the impact of job role on rating a company using the overall-ratings field by the company by year.

Display the relationship between the overall rating score vs. the rest of the rating field scores by company. Also, document your findings.

- Overall-ratings

Versus

- Work-balance stars
- Culture values stars
- Career opportunities-stars
- Comp-benefit-stars
- Senior-management-stars

Document your findings for the following:

- a) Which corporation is worth working for
- b) Classification of satisfied or unsatisfied employees

```
create database reviews;
use reviews;
```

OK

```
CREATE EXTERNAL TABLE  reviews.NP_review_data (
  Index INT,
  company STRING,
  location STRUCT<city:STRING,country:STRING>,
  dates STRUCT<dor:STRING,yor:STRING>,
  `job-title` STRING,
  summary STRING,
  pros STRING,
  cons STRING,
  `overall-ratings` INT,
  `work-balance-stars` INT,
  `culture-values-stars` INT,
  `carrer-opportunities-stars` INT,
  `comp-benefit-stars` INT,
  `senior-management-stars` INT)
row format delimited
fields terminated by ','
collection items terminated by ';'
TBLPROPERTIES ("skip.header.line.count"="1")
LOCATION 'dbfs:/FileStore/shared_uploads/faizalnajeeb761@gmail.com/Review';
```

OK

```
load data inpath 'dbfs:/FileStore/shared_uploads/faizalnajeeb761@gmail.com/Review_data/employee_review_data.csv' into
table NP_review_data;
```

OK

```
CREATE EXTERNAL TABLE  reviews.DP_review_data (
  Index INT,
  company STRING,
  years STRING,
  `job-title` STRING,
  summary STRING,
  pros STRING,
  cons STRING,
  `overall-ratings` INT,
  `work-balance-stars` INT,
  `culture-values-stars` INT,
  `carrer-opportunities-stars` INT,
  `comp-benefit-stars` INT,
  `senior-management-stars` INT)
partitioned by (country STRING)
clustered by (years) into 10 buckets
row format delimited
fields terminated by ','
TBLPROPERTIES ("skip.header.line.count"="1")
LOCATION 'dbfs:/FileStore/shared_uploads/faizalnajeeb761@gmail.com/Review';
```

OK

```
set hive.exec.dynamic.partition.mode=nonstrict;
```

Table

	key	value
1	hive.exec.dynamic.partition.mode	nonstrict

Showing 1 row.

```
insert into table reviews.DP_review_data partition (country)
select  Index ,
  company,
  dates.yor,
   `job-title`,
  summary,
  pros,
  cons ,
  `overall-ratings`,
  `work-balance-stars`,
  `culture-values-stars`,
  `carrer-opportunities-stars`,
  `comp-benefit-stars`  ,
  `senior-management-stars`,
  location.country
from reviews.NP_review_data;
```

OK

```
desc extended reviews.DP_review_data;
```

Table

	col_name	data_type	comment
1	Index	int	null
2	company	string	null
3	years	string	null
4	job-title	string	null
5	summary	string	null
6	pros	string	null
7	cons	string	null

Showing all 37 rows.

```
select `overall-ratings`,count(*) from reviews.DP_review_data
group by `overall-ratings`;
```

Table

	overall-ratings	count(1)
1	1	56850
2	3	157974
3	5	251325
4	4	250116
5	2	72603
6	null	21492

Showing all 6 rows.

Using the over-all rating fields display trend for company:

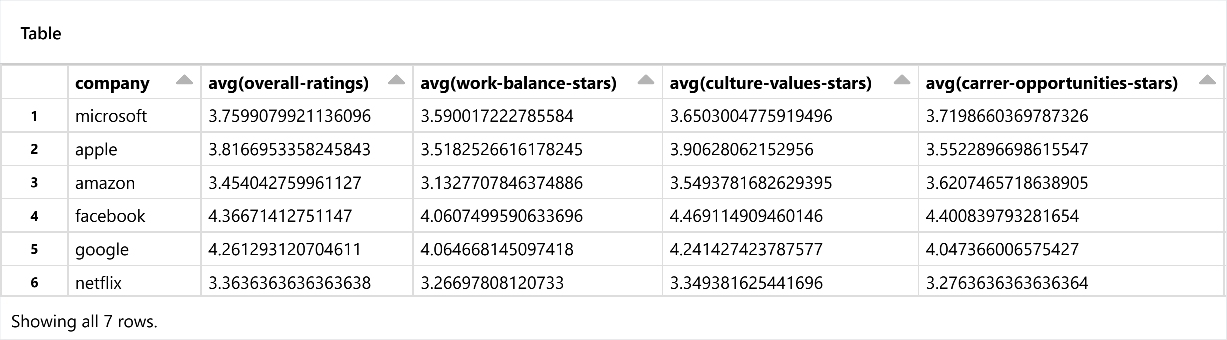
- country wise
- year wise

```
select distinct t.* from
(SELECT company,country, NTILE(4) OVER (ORDER BY `overall-ratings`) AS country_quartile
FROM reviews.dp_review_data) t
;
```

```
select distinct t.* from
(SELECT company,years, NTILE(4) OVER (ORDER BY `overall-ratings`) AS year_quartile
FROM reviews.dp_review_data) t
;
```

Overall rating score vs. the rest of the rating field scores by company

```
select company,
  avg(`overall-ratings`),
  avg(`work-balance-stars` ),
  avg(`culture-values-stars`),
  avg(`carrer-opportunities-stars`),
  avg(`comp-benefit-stars` ),
  avg(`senior-management-stars`)
from reviews.dp_review_data
group by company;
```



Table

	company	avg(overall-ratings)	avg(work-balance-stars)	avg(culture-values-stars)	avg(carrer-opportunities-stars)
1	microsoft	3.7599079921136096	3.590017222785584	3.6503004775919496	3.7198660369787326
2	apple	3.8166953358245843	3.5182526616178245	3.90628062152956	3.5522896698615547
3	amazon	3.454042759961127	3.1327707846374886	3.5493781682629395	3.6207465718638905
4	facebook	4.36671412751147	4.0607499590633696	4.469114909460146	4.400839793281654
5	google	4.261293120704611	4.064668145097418	4.241427423787577	4.047366006575427
6	netflix	3.3636363636363638	3.26697808120733	3.349381625441696	3.2763636363636364

Showing all 7 rows.

Complimentary benifits is affecting overall ratings negatively for all companies while senior management ratings are affecting overall ratings positively

Facebook overall ratings are good compared to other so it is worth joining for most employees.

Ratio of satisfied to unsatisfied employees is high > 3.4 on average