Extension for Analyzing Employee Satisfaction Trend

Name: Huijun Lin

Leader / coordinator: Huijun Lin (One person group)

NetID: huijunl2

Email: huijunl2@illinois.edu

Subject Areas: Free Topics

1. Overview

In this project, I will write a chrome browser extension that analyzes the company employee satisfaction trend (CEST).

2. Background

The CEST extension relies on employees' text reviews to analyze the company employee satisfaction trend. It crawls the review data from the Glassdoor website and analyzes it with a sentiment analysis tool to get a time chart on how the company's employee attitudes changed over time (i.e. monthly / yearly). According to the chart, users can intuitively feel the changes in the satisfaction and confidence of the internal personnel of the enterprise. Compared with outsiders, internal stakeholders often have more information about the development of the enterprise, so this curve can help them understand the development of the enterprise more clearly and guide them to evaluate the enterprise.

2.1 Target users

shareholders, investors, job seekers, employees, company owners.

2.2 Demand Analysis

Nowadays, if people want to learn or predict the development of a company, they usually rely on external data, such as stock price, news, company associates, etc. These information do provide some knowledge to help company evaluations, but also have a few drawbacks:

1. The acquisition time is scattered.
2. External information is untrustworthy.
3. Most of them are enterprise dynamics related, and are barely about internal management.

In view of the above shortcomings, people will encounter the following problems when analyzing these data:

1. Difficulty spotting trends over time.
2. The authenticity is questionable, and the observation of internal changes usually delays.
3. You can only see the "development trends" that the company is willing to expose, but do not understand the internal management and operation status

Almost all of the evaluation data in Glassdoor comes from current or former employees of the company. Employees tend to have more insider truth than outsiders. Sentiment analysis of these evaluations can reflect employees' satisfaction with the company and confidence in future development. Moreover, these data have time continuity, and through data visualization, a curve of employee attitudes changing over time can be drawn. In this way, users will obtain temporally comparable internal attitude information, which will guide users to evaluate a company more objectively and comprehensively.

In addition, although each user review on Glassdoor comes with a rating and a curve can be drawn from them, since these data have not been normalized and thus have limited reference value. For example, people that are easy to satisfy usually give 5 stars, and people that are more strict usually do not give 5 stars unless extremely impressed). By using sentiment analysis tools to analyze all reviews over a period of time, we can convert reviews into data with consistent standards, which is more conducive to showing overall attitude changes.

After researching, I found there are no existing tools that have the same function. Therefore, I think users can benefit from the tool I am building.

3. Implementation Details

On the high level, I will build an extension with the core user journey being:

As a user viewing a Glassdoor company page, clicking the extension button will display a popup showing the normalized employee satisfaction time chart divided by month/year.

3.1 Extension Components

In order to meet the requirement above, the extension can be divided into 3 components:

1. Crawler: fetch the sampling or all the reviews given a Glassdoor company page.
2. Analyzer: Map all the reviews using sentiment analysis into normalized scores.
3. Visualizer: Plot all the data with x-axis being the time and y-axis being the score.

3.2 Milestones / Road Map

Here is the high level implementation plan:

Plan	ETA
Build a “hello world” chrome extension using the provided APIs.	3 hours
Build the Crawler component	6 hours
Build the Analyzer component	7 hours
Build the Visualizer component	4 hours

3.3 Challenges

1. Crawler
   - I might need to learn and apply different levels of crawler resistance based on Glassdoor’s security setup.
2. Analyzer
   - Tensorflow is a complicated library and there are many different ways to achieve a goal. I will learn it by doing it.
   - I need to investigate and experiment with a data set that can yield my expected results.
3. Visualizer
   - I need to learn how to draw charts using javascript libraries such as D3.js.

3.4 Specifications

1. Programming language: Javascript
2. Tools: Chrome Extension API, Tensorflow JS (tfjs), D3.js (May vary along with progress)
3. Datasets: I will investigate and experiment different trained models based on my usage from tensoflow’s datasets.

3.5 Requirements / Expectations

1. The extension should be able to run for different company pages in Glassdoor.
2. The extension should be able to plot each data point with the coordinate (time, score).

3.6 Evaluation

1. Test different companies and display different results (assuming none of the companies will have the same result).
2. Evaluate the Crawler component by checking the targeted reviews are fetched.
3. Evaluate the Analyzer component by checking whether the returned score for a given review is reasonable (involves human effort). Evaluate both positive & negative reviews.

4. Evaluate the Visualizer component by checking whether the rendered time chart matches the reviews.

References

Google Developers. (n.d.)*. API Reference*. Retrieved October 23, 2022, from https://developer.chrome.com/docs/extensions/reference/

TensorFlow. (n.d.). *TensorFlow.js Example: Sentiment Analysis*. GitHub. Retrieved October 23, 2022, from https://github.com/tensorflow/tfjs-examples/tree/master/sentiment

TensorFlow. (n.d.). *TensorFlow Datasets: a collection of ready-to-use datasets*. Retrieved October 23, 2022, from https://www.tensorflow.org/datasets

Bostock, M. (2013). *D3.js - Data-Driven Documents*. D3js.org. Retrieved October 23, 2022, from https://d3js.org/