Recalibrated Restaurant Star Ratings with LLMs

Sang-won Shim

Presenter Introduction: Sang-won Shim

Social Media Links

LinkedIn
Profile

Capstone
GitHub

Presentation
Deck

Online reviews and ratings can make or break a business

88% of consumers **read online reviews to decide** whether to experience or purchase a business' product or service [1]

94% of consumers **will not do a business** with a company due to **negative online reviews** [1]

Which restaurant do you want?

OR

[1] Data Source: Digital Air Strike

Business Problem

Current Challenge

➔ Restaurant owners believe Google Maps reviews are **arbitrary and inconsistent**

Project Scope

➔ Investigate whether the **user complaints are valid** so that Google Maps PM can make a data-driven decision

Note: This business problem is hypothetical

Data Overview

Source

➔ Google Maps API

Data

➔ **~3K star ratings and text reviews** for ~570 unique restaurants

Limitations

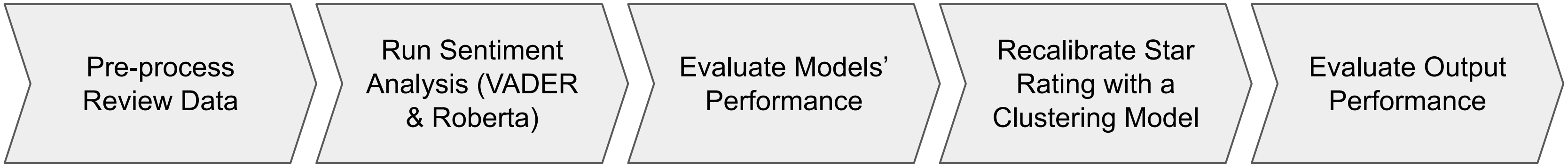
➔ Due to the limited computational power, a **small dataset** is used for the analysis

➔ The dataset does not have all/full text reviews for a given restaurant (i.e., limited to **5 reviews per restaurant**)

➔ The dataset exclusively examines reviews of **New York City restaurants**. Different geography may have quite different relationship between text review's sentiment score and star rating

Analysis Process Overview

Analysis Process



Pre-process Review Data

Run Sentiment Analysis (VADER & Roberta)

Evaluate Models' Performance

Recalibrate Star Rating with a Clustering Model

Evaluate Output Performance

LLM / Sentiment Analysis Model Used

- ➔ VADER
- ➔ Roberta (latest model from 2022)

Clustering Model Used

- ➔ K-Means
- ➔ Agglomerative

Evaluation Method

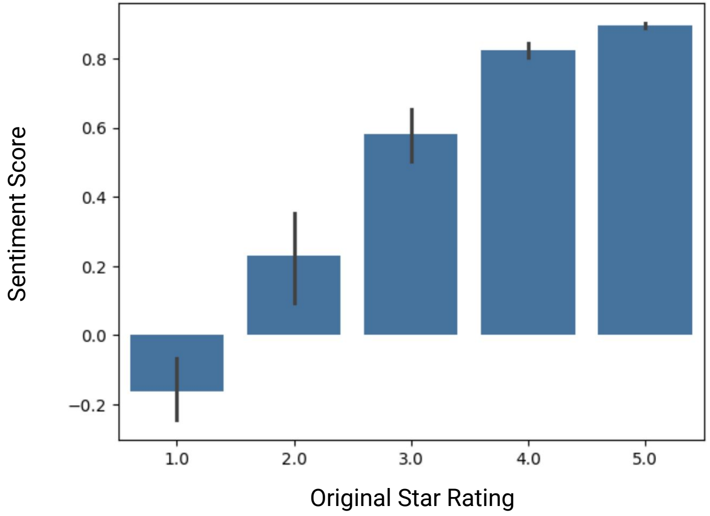
- ➔ Sentiment Scores
- ➔ Human Inspections

Important Note

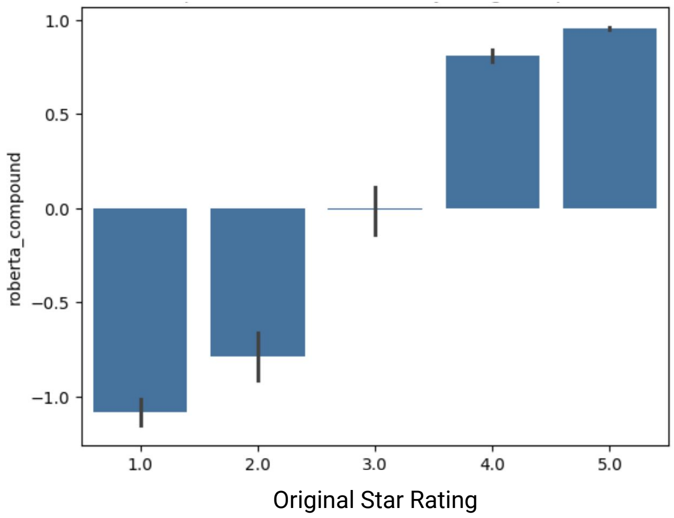
- ➔ Sentiment analysis model does not require usual NLP preprocessing steps (e.g., tokenizing, stemming, and lemmatizing) as the models rely on **pre-built lexicon** that contains sentiment scores

At the aggregated level, the current star rating system/bucket appears to be relatively accurate

VADER Compound Sentiment Score

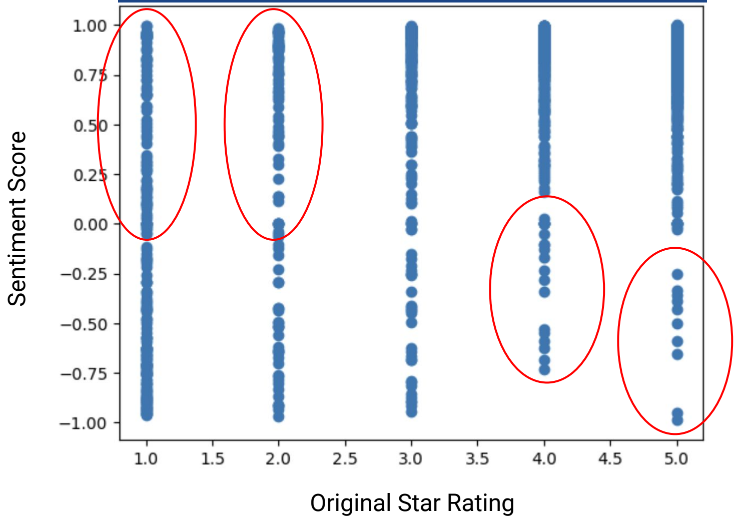


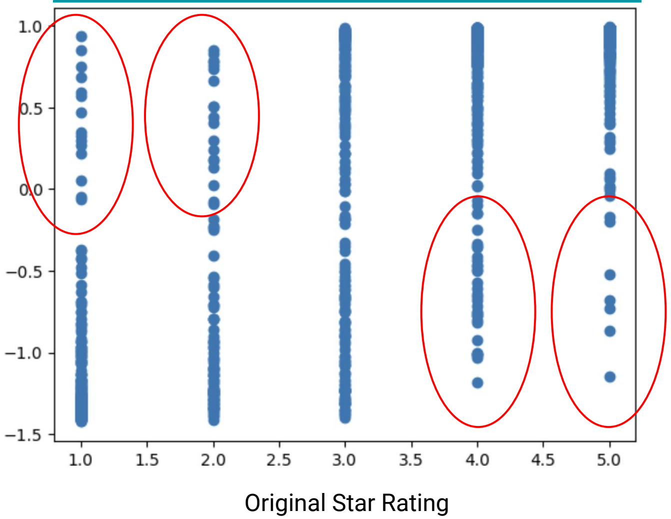
Roberta Compound Sentiment Score



➔ **On the aggregated level, there is a clear distinction** between each star rating and average compound sentiment score for both models

At an individual review level, there are many instances where star ratings and text reviews were mismatched


VADER Compound Sentiment Score


Roberta Compound Sentiment Score

- ➔ On an individual review level, there is a **wide variation in sentiment score** for a given star rating
- ➔ There must be some **mismatch** between star ratings and text reviews

Both LLMs sometimes made mistakes with nuanced and subtle reviews

Selected 1-star Review
“You’re paying $9 for the fancy truck to get ripped off with mediocre food, barely any chicken and a ton of raw onions after on top to create an illusion. The guy also shouting talking on phone preparing your food spit flying out his mouth. So many better other options near. Never again”

	Negative	Neutral	Positive
VADER	0.000	0.874	0.126
Roberta	0.946	0.048	0.005

Another 1-star Review
“Delicious dumplings were only surpassed by the warm beer. Limit your ordering to that.”

	Negative	Neutral	Positive
VADER	0.073	0.615	0.313
Roberta	0.013	0.053	0.932

- ➔ Based on several spot checks, **Roberta model appears to have better performance** than VADER model
- ➔ However, neither LLMs were good at picking up on **context or nuances**

The final output correctly re-calibrated star ratings in many instances, but it was not perfect

Example of Good Re-classification (Disagree with user’s 3 star rating, and it was re-calibrated as 5 star review)

"Loved their pizza and calzones! We loved how the pizza bread were chewy and had some garlicky flavor, slices were big and affordable too. Their menu options is wide and freshly made! Will definitely go back when I'm in the city."

Example of Mis-classification (Agree with user’s 3 star rating, but it was re-calibrated as 5 star review)

'First off this place is smaller than it looks. The tables are tiny. The pizza is OK. It is not amazing but not awful. The ingredients are fresh and tasty. The service is great here. Prices are good. The place is pretty and stylish.'

➔ Since the LLM models were not great at picking up on contexts, nuanced reviews, and sarcasms, clustering performance was **not always optimal**

Recommendations for the PM

1 **Provide Guidance**

Post **example reviews** per different star rating buckets to create a general criteria/guideline for what each star rating means (to prevent very harsh or loose star ratings)

2 **LLM Based Star Rating**

Beta-test **LLM-based restaurant star rating system** (i.e., based on sentiment analysis model) in addition to the traditional star rating system and get user's feedback

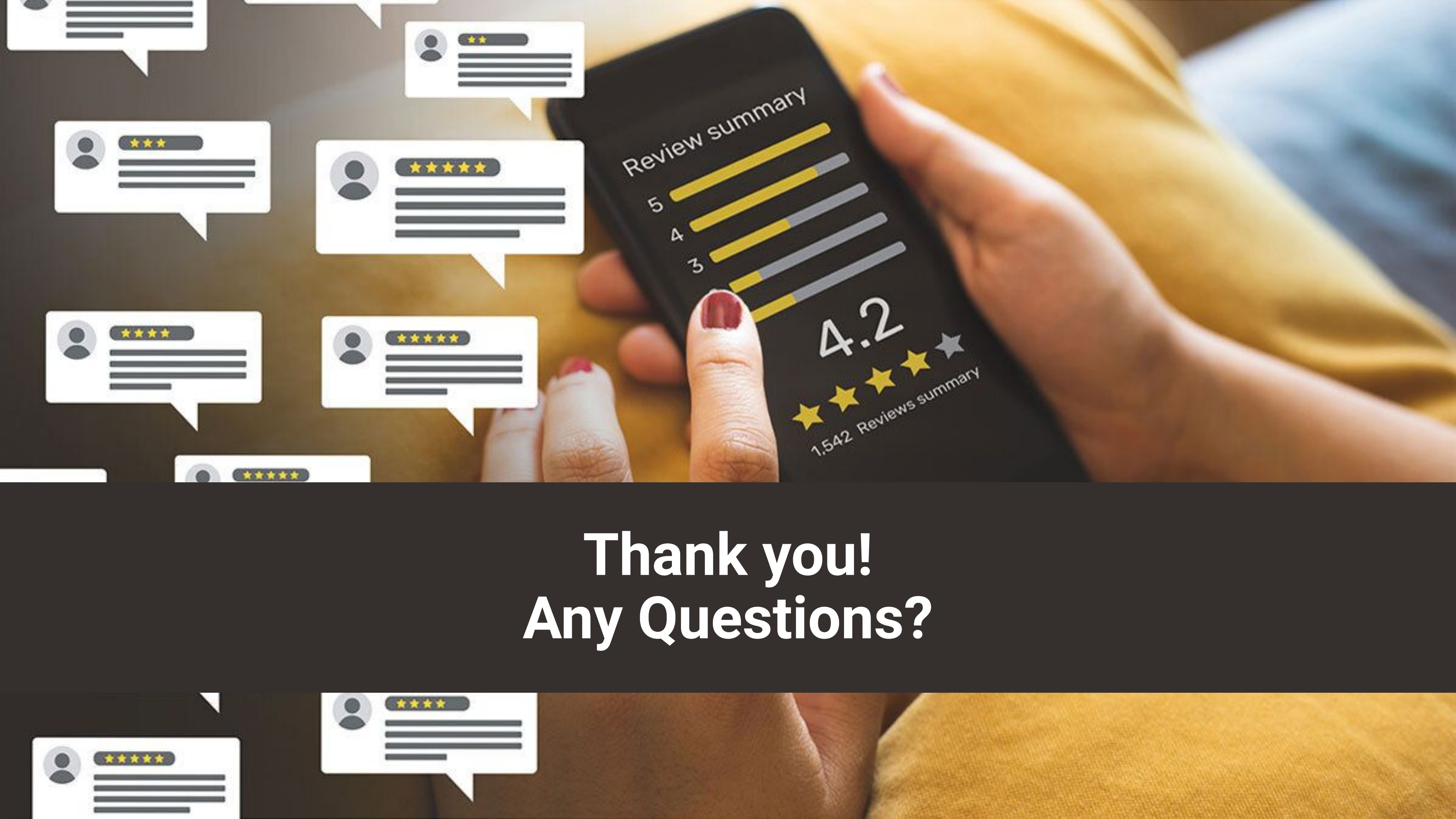
3 **Assistant Feature**

Beta-test **star rating assistant feature**: when there appears to be a big mismatch in text review's sentiment score and user's star rating, assistant feature can provide a suggested star rating

With the recommendations, the PM should be able to address concerns from the restaurant owners

Next Steps

- ➔ With more computational power, re-run the analysis with a bigger dataset
- ➔ Research into more advanced sentiment analysis model that can better pick up nuances (e.g., ChatGPT 4.0 API)



Thank you!
Any Questions?