Comprehensive Rank Strategy for Yelp Customer Reviews

SI671 | Xiaoyang Sheng, Yulin Gao, Zicong Xiao

Background

• E-commerce, e.g., Yelp, faces challenges in managing customer reviews' impact on sales and reputation.
• Relevant reviews positively boost sales, while deceptive ones can bring financial losses and even erode customer trust.
• Our goal is to develop an effective strategy to rank and present the reviews.

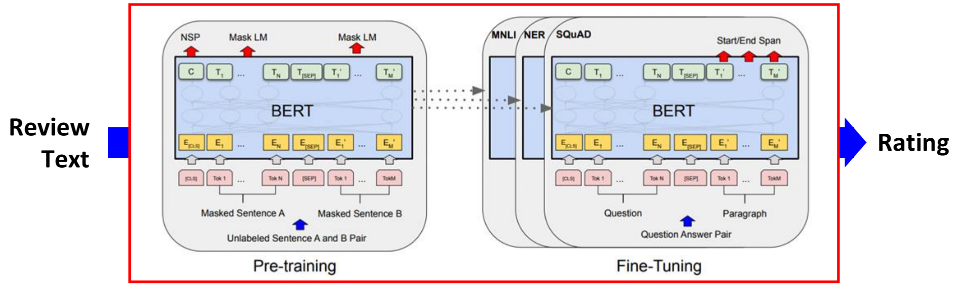
Data

• Yelp's dataset contains comprehensive business information, including location and attributes [1].
• It also includes **full review text** with the associated user and business IDs, as well as **user profile info** like friend mappings.

RQ2/M2. Predict if review rating match the text

• Rating from a review falls in the range of [0, 5] stars. We utilized the pre-trained BERT, an open-source distilled multilingual version from Hugging Face community. On average, the Distil-mBERT (134 parameters), is twice as fast as normal mBERT (177 parameters). [2] [3]
• To predict rating from review text in a regression setting, we customized the Adam optimizer as well as the categorical cross-entropy loss function, and added another extra layer to the end, computing a weighted average of each rating (in star).

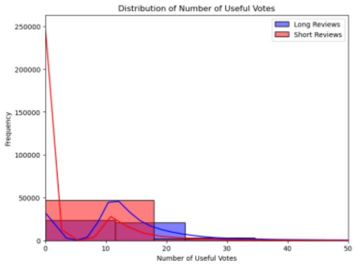
(MAE = 0.90)

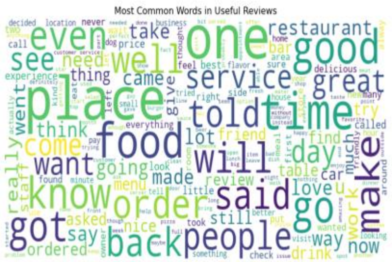


RQ1/M1. Predict if the review text useful

• Usefulness is quantified by the number of ***useful*** votes received. **"Useful" cat.** if the number is greater than 10, otherwise **"useless"**.
• Distribution of "useful" reviews differs a lot as the overall length or common words in review text vary. BERT classifier is introduced to divide upcoming reviews into the two cats by **sequential review text**.

(Validation ACC. = 0.79)

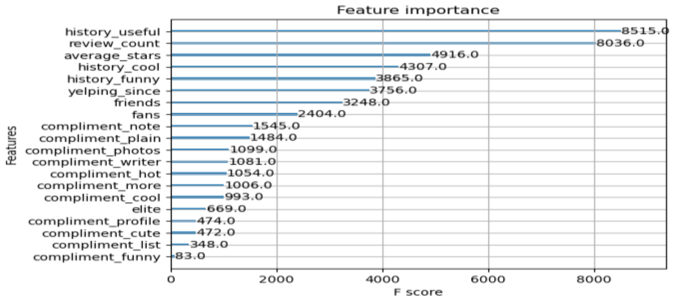




RQ3/M3. Predict how useful the review can be

• As it takes time to accumulate ***useful*** votes, the potential (numeric) usefulness of an upcoming review can be predicted based on the reviewers' profile information, again, regression.
• XGB regressor is well-tuned by GSCV on 20 features as vector. Most helpful features are ***history_useful*, *review_count*, *average_stars* ...**

(MAE = 5.40, std(useful) = 14)



Ranking Strategy

0. New Review & User Info (Input) → 1. (M1) Predict if useless review → 2. (M2) Predict if outlier rating → 3. (M3) Predict potential usefulness → Rank by (rating, model score, pub. date) DESC

• Review
{ Rating, Text }
• User
{ review_count, history_useful, yelping_since, ... }

If YES, give 0 score

If YES, give 0 score

Pred. score as model score

M. Score	Review Text (Sample for Rating = 5.0 star)
10.88	This is the best cheesesteak in the city hands down. And it has been for a long time. Even before it was called joes. Everything ...
9.74	CHOCOLATE FONDUE FOUNTAIN. Need I say more?! Well. They have strawberries, marshmallows, and wafers you can dip into ...
...	
0	I can say this is the best burger of my life!

Limitations

Current methods only consider information from words in review text using NLP models. However, the younger generation may prefer to use emojis 😄 to vividly convey their emotions, where our current methods can have poor performance.

Reference

[1] Yelp. Dataset. Accessible at www.yelp.com/dataset. (2023).
[2] Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding.
[3] Sanh, V., Debut, L., Chaumond, J., & Wolf, T. (2019). DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter.