First, I admit that you did a lot of works for using some advanced models which I did not cover in this course. However, I think this report needs additional efforts in logics and writing.

The first round review score would be:
70(Base)+2(Writing)+2(logic)+3(difficulty)
=77.

Looking forward to your revision.

Add your UID and departments of your majors here

rant Review Challenge

Alan Wang, Hongyi Yang, Jiayu Hu

9 December 2022

This reports considers the rating prediction problem based on the Yelp challenge dataset.Then (more details how you do)

Abstract

Our group built a user rating predictor using machine learning models. Beginning with a simple baseline model, we further explored modeling methods such as linear regression and random forest. In addition, we perform ~~cross validation~~ to find the optimal hyperparameters ~~in increasing efficiency or decreasing the margin of error.~~ In the end, we implement more high-end model including bag-of-bigrams and BERT with deep learning. Overall, although the dataset is highly imbalanced, ~~for convenience~~, we ~~will~~ still utilize each of our models on the test set and use accuracy and r^2 as the metrics of the performance of each model.

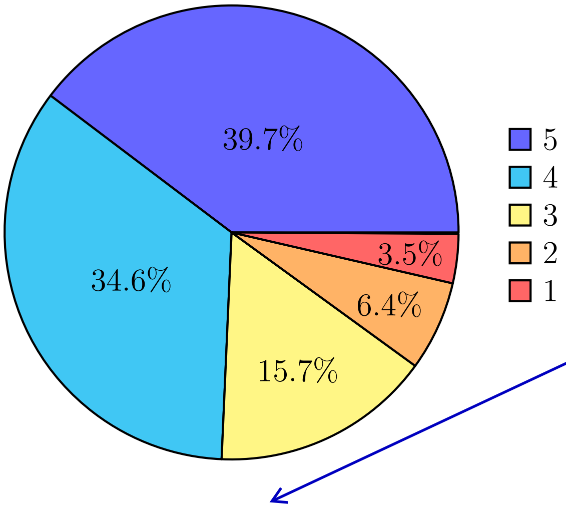
1 Background

Yelp has been one of the most popular Internet rating and review sites for local businesses since its initial inception in 2004. It is an online review site in which customers ~~shared~~ their experiences, helping others make informed decisions about restaurants, auto-repair shops, and more. Those reviews allow consumers to trust a small business and be confident in their purchasing decisions. From a ~~small~~ business perspective, the higher the review ratings on Yelp, the better chance it will attract more consumers.

2 Dataset

The provided dataset is on the customers' reviews of different restaurants in California. Given a review, the dataset includes the customer's rating (1-5), text comments, ~~others'~~ reactions toward the commenting, and some information about the user. The dataset also contains geographical information on the restaurants.

~~The data is highly unbalanced. We make a piechart of the distribution of the ratings:~~



You should add index for figure. This should be Figure 1.

We determine~~d~~ that our goal is to predict one user's rating from the corresponding text comment. Since the ratings are nominal, we can proceed with either a classification approach or a regression approach, and we will experiment with both in our modeling.

3 Data Preprocessing

The data preprocessing step should contains the part how you get your predictors from textual data. Specify what kinds of features you final put in your algorithm.Removing unnecessary characters is just one setp.

Before modeling start, a series of text clearing will be necessary to ensure the quality of data. In our sentiment analysis, we want to remove the unnecessary punctuation by lambda expressions and regular expressions, and ~~remove~~ "stopwords", which is a collection of words that are "neutral" and ~~very common/everyday use~~ words, just so we can trim out the unnecessary characters that make our dataset too long to compute. Then, we remove "punctuation," which we pause, stop, emphasize, or question using a comma, a period, an exclamation point or a question mark. Furthermore, we remove all non-English characters and change all the letters into lower case. Finally, we delete all the numbers and excessive spaces and return lines.

We divide the dataset to a training set and a testing set on a 7:3 ratio.

Merge them together

4 Modeling

4.1 Baseline Model

~~A baseline model is a simple model to which we compare the performance of our future complicated models. If our trained models are unable to outperform the baseline model,~~

~~it could be a sign that the data set lacks predictive power. We use the mean rating of the training and set it as the prediction of the test set, regardless of the review text~~. The baseline yields an r^2 of 0, ✓

4.2 TF-IDF + Linear Regression

The TF-IDF is the product of two statistics, term frequency and inverse document frequency. Here, term frequency, $tf(t,d)$ is the relative frequency of term t within document d,

$$\mathrm{tf}(t,d) = \frac{f_{t,d}}{\sum_{t' \in d} f_{t',d}}$$

comma should be within the equation.

~~,~~ where $f_{t,d}$ is the raw count of a term in a document. In our example, we use the frequency that each non-stopword term appears in each review. Linear regression is a linear approach for modeling the relationship between a scalar response (dependent variable and one or more explanatory variables (independent variables) with formulation

$$\mathbf{y} = \mathbf{X}\beta + \epsilon$$

missing a comma

where $\mathbf{y}$ is the dependent variable, $\mathbf{X}$ is the design matrix of predictor variables, β is a $(p+1)$-dimensional parameter vector with β_0 the intercept term (if one is included in the model—otherwise β is p-dimensional), and **epsilon** is the error term. ~~We wish to minimize the error term $\epsilon = \mathbf{y} - \mathbf{X}\beta$ when estimating the coefficient vector β to fit the model.~~ In the case of this project, we take the number of stars given by an individual user to a single restaurant as our dependent variable $\mathbf{y}$, the TF-IDF statistics for each review as predictor $\mathbf{X}$. This model yields an r^2 of -40.00 and a mean squared error of 46.53. It achieves an accuracy of 29.60%.

I am not sure whether the R^2 you define here is the R^2 I know. Basically, R^2 is always positive. Additionally, I think a possible reason why you have a high R^2 for a regression model is that you did not normalize your features. Since the magnitudes of TF could be highly varying from document to documents. Another thing is that how a linear regression model can have a accuracy. Could you specify how you transform the continuous prediction to categorical response.

4.3 Sentiment Le

✓ A sentiment lexicon is a collection of words (also known as polar or opinion words) associated with their sentiment orientation, that is, positive or negative. [2][3] Random forest is an ensemble learning method that operates by a multitude of decision trees, the output of which is determined by the selection of the most trees. To train the model, we combine matrix produced by applying sentiment lexicon on the training set with random forest. Note that ~~since~~ random forest already corrects for decision trees' habit of overfitting to their training set, ~~we don't need~~ ^ to employ cross-validation in this model. This model yields an r^2 of 0.08 and a mean squared error of 1.04. It achieves an accuracy of 49.55%.

usually references will not be put before the setences.

4.4 Bag-of-Bigrams Regression

In a big-of-bigram model, instead only look at single words, we also look at groups of 2 consecutive words before applying a text vectorizer. This is a set model, meaning we discard order or the words and treat text as an unordered set of words. This makes sense because when we try to predict the customer's rating on restaurants, we look for keywords such as "great restaurants" or "terrible" instead of their relative order. We feed the vectorized text into a simple neural network and let it do the trick for us.

We apply a simple neural network: a layer with 16 nodes with Relu activation, a dropout layer, and a final layer for output. We also limit the maximum and minimum to 5 and 1, respectively.

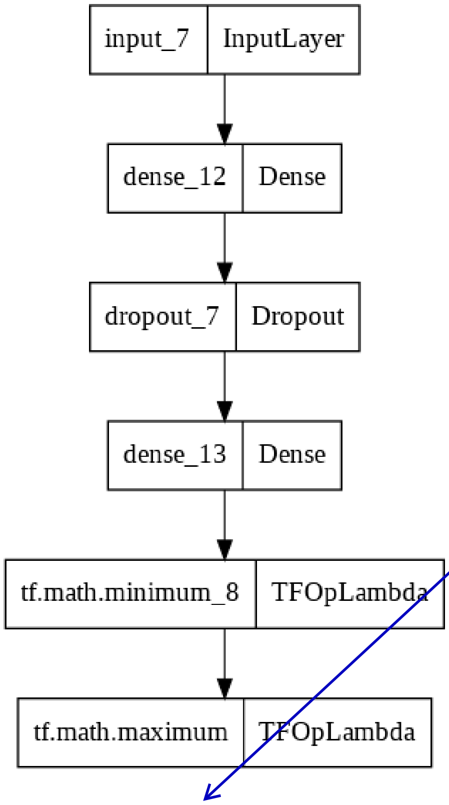


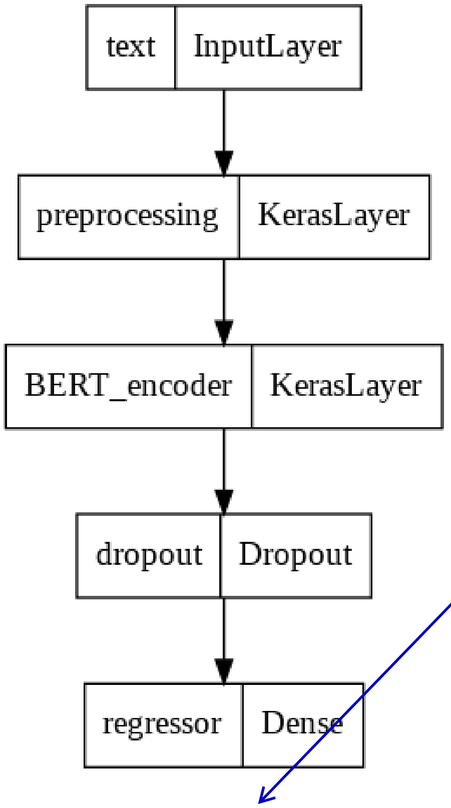
Figure index

Our model achieves an accuracy of 20% and r^2 value of -0.44.

4.5 BERT Classification

BERT is short for Bidirectional Encoder Representations from Transformers. The idea of Transformers comes from the famous paper "Attention Is All you Need" [4], where the authors proposes a revolutionary transformer system that solely relies on attention mechanisms to learn global dependencies between input and output and overcomes the parallelization problem often seen in Recurrent Neural Network models. Google's BERT model's is a pretrained unsupervised model trained on millions of English text; ~~unlike the set model in Section 4.4~~; BERT provides contextualized embedding for words: embedding might be different for the same word according to the sentence. For our project, we uses Google's BERT Small Uncased model [1].

For our model, we add a Dropout layer of 0.1 to prevent overfitting, and a ~~dense~~ layer of 5 with ~~sigmoid activation function~~ ~~for classification purpose~~. ~~Overall, our model looks like this:~~



Add figure index

Our model achieves a 60% accuracy on test set with a r^2 score of 0.48, very high given the randomness in the nature of text data.

5 Results and Discussion

Model	**Accuracy**	r^2
Baseline Model	0.346	0
TF-IDF + Linear Regression	0.296	-40.00
Sentiment Lexicon + Random Forest	0.496	0.08
Bag-of-Bigrams Regression	0.200	-0.44
BERT Classification	0.595	0.49

There are three things we noticed: 1. BERT being the current state-of-art model, undoubtedly achieves the best result. 2. ~~Classification overall achieves a much better performance than regression.~~ Although the data is nominal, it is not continuous, causing the issue. If we the change the what we did in Section 4.4 to classification, it should yield a better performance. 3. Overall the ratings for Californian restaurants are very high, with an average of 4 out of 5. It could be a result of Yelp users in this area being generous in giving stars or the quality of food and service being genuinely satisfying.

References

[1] Turc, Iulia and Chang, Ming-Wei and Lee, Kenton and Toutanova, Kristina. “Well-Read Students Learn Better: On the Importance of Pre-training Compact Models”. *arXiv preprint arXiv:1908.08962v2*. 2019.

[2] Minqing Hu and Bing Liu. ”Mining and Summarizing Customer Reviews.” *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD-2004)*, Aug 22-25, 2004, Seattle, Washington, USA

[3] Bing Liu, Minqing Hu and Junsheng Cheng. ”Opinion Observer: Analyzing and Comparing Opinions on the Web.” *Proceedings of the 14th International World Wide Web conference (WWW-2005)*, May 10-14, 2005, Chiba, Japan.

[4] Ashish Vaswani, et al. ”Attention is All You Need.” *ArXiv, arXiv:1706.03762v5*, June 12, 2017.