CSC 180-01 Intelligent Systems (Fall 2024)

Title: - Yelp Business Rating Prediction using TensorFlow

Due at 10:30 am- Wednesday, September 25, 2024

Name	**Student ID**
Taekjin Jung	303293432
Illya Gordyy	302682939
Jenil Shingala	302796429
Danny Phan	301698774

1. Problem Statement

In this project, we built a neural network to predict star ratings for businesses on Yelp based on their reviews. Our main challenge was to convert the text of reviews into dummy columns using One-hot encoding and TF-IDF that our neural network could understand and use to make accurate predictions. We treated this as a regression problem, where our network needs to output a single number (the predicted star rating) based on the input of review text. This project helped us learn how to process text data, build and train neural networks, and evaluate their performance on real-world data

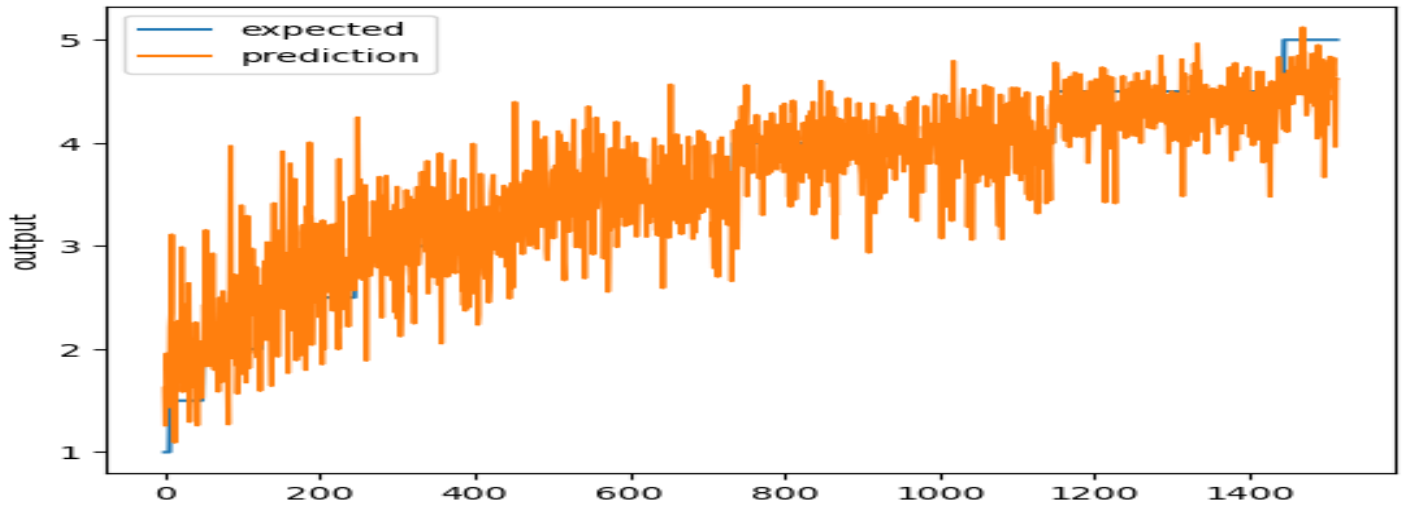
2. Methodology

We used the Yelp dataset for our project, focusing on the business and review data. First, we cleaned the data by keeping only businesses with at least 20 reviews and picking the random 10000 business to choose. This helped ensure we had enough information about each business. To prepare the review text for our neural network, we used a method called TF-IDF (Term Frequency-Inverse Document Frequency). This turned the words in each review into numbers that show how important each word is and we discarded the common words like the, is, and gave the minimum 10 percent and then maximum 90 percent . We split our codes into three parts: dealing with data sets of the review.json and business.json by merging, filtering, normalizing, and using TF-IDF, training/testing the model, and a sample test set to check how well it learned. We built our neural network using TensorFlow. Our network had several layers of neurons(64/32/1), with each layer using a function (like ReLU) to process its inputs. We tried different network structures, changing the number of layers and neurons. We also tested different optimization methods (Adam and SGD) to help our network learn

better. To prevent our network from memorizing the training data instead of learning general patterns, we used a technique called EarlyStopping

3. Experimental Results and Analysis

	Activation	Layers and neuron counts	Optimizer	RMSE
Model 1	Tanh	64/32/1	Adam	0.3900589942932129.
Model 2	Sigmoid	64/32/1	Adam	0.374832272529602.
Model 3	Relu	64/32/1	Adam	0.4527590870857239.
Model 4	Relu	100/10/1	Adam	0.467869400978088.
Model 5	Sigmoid	64/32/1	Sgd	0.4002636671066284.



As a result, we got the best RMSE:0.374832272 with a model trained by sigmoid (activation), 64/32/1(number of neurons), and adam(optimizer). The second figure is the lift chart with the best regression model.

4. Task Division and Project Reflection

Name	Task
Taekjin Jung	Data management, train/test model, visualization.
Illya Gordyy	Data management, tf-idf, train/test model.
Jenil Shingala	Data management, testing different hyperparameters, report
Danny Phan	Data management, Debugging

Challenges:

- Hardship with TF-IDF Vectorizer (no experience)
    - Referred the library
- Database management (dropping, merging, grouping, sorting)
    - Discussed with teammates to figure out the solution
- Time management
    - Planning and spending a decent amount of time

Learning Outcome:

- How to use TF-IDF Vectorizer
- How to manage data sets efficiently
- Better task distribution
- Applying the knowledge of the lectures and labs