Sentiment Analysis on Amazon Review

Wang Haonan 3036034576,

Wang Junheng 3036032530,

Zhang Weiwen 3036034734

Faculty of Engineering, the University of Hong Kong

Comp 7103 Assignment 1 Final Project Report

April 18, 2023

1. Introduction

The growth of e-commerce has led to an unprecedented increase in the number of online reviews. Reviews on leading e-commerce platforms like Amazon and e-Bay can influence consumers' purchasing decisions deterministically. Understanding the sentiment expressed in these reviews can provide valuable insights to businesses, enabling them to improve their products and services.

Sentiment analysis is a branch of natural language processing that uses machine learning algorithms to identify and extract subjective information from text. One of the most popular methods is to classify text sentiment into three polarity categories: positive, negative, and neutral. In addition to the technique above, our goal was to establish an architecture that could accurately classify Amazon reviews into five different polarity levels based on each review's text content and its helpfulness votes from other users. In this project, we have performed a series of sentiment analysis on Amazon reviews to predict the score related to the text and compare our prediction to the real rating scores given by users.

Our motivation for this project was to explore the potential of sentiment analysis as a tool for businesses to gain insights into customer feedback. By analyzing the data and metadata on the Amazon reviews we have collected, we aimed to identify trends and patterns in customer sentiment towards different products. Our results can be used to inform decision-making in areas such as product development, marketing, and customer service. Our code is available at **https://github.com/Caffretro/Comp7103_Assignment1_data_mining_and_NLP.**

2. Data Preprocessing

The review data we have collected consists of more than 10 columns[1]. We believed using data visualization techniques would help us better understand the distribution and excavate hidden correlations within the dataset, supporting the establishment of our models. Here are some of the graphs we have plotted:

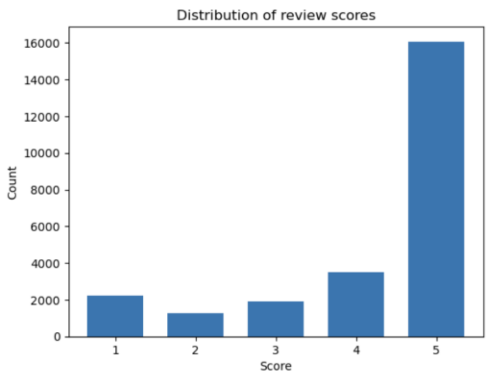

Figure 1. Histogram of Scores. Most users are very satisfied with the product they

[1] https://www.kaggle.com/datasets/snap/amazon-fine-food-reviews

bought; users who rated for 1-star are twice the number of users who rated 2-stars.

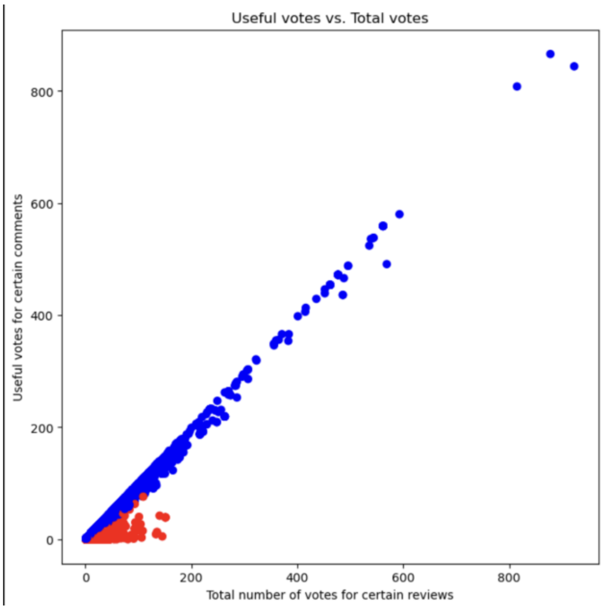

Figure 2. Scatter Plot of Debatable Usefulness. If we set 70% as the threshold for 'debatable usefulness', we can see that reviews that have received less than 70% useful votes tend to gather fewer total votes.

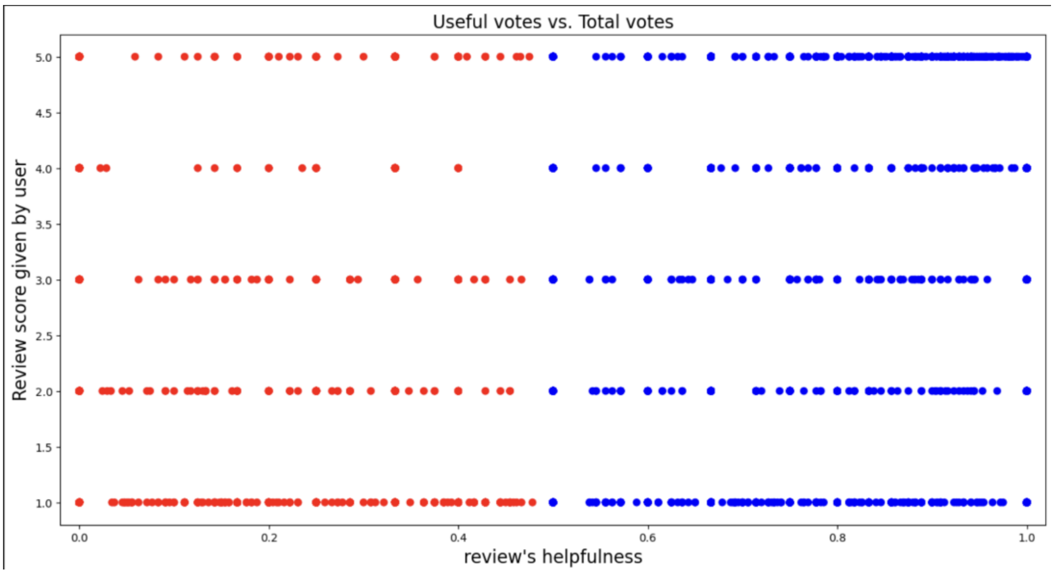

Figure 3. Scatter Distribution of Helpfulness. For user ratings above 2-stars, other users tend to vote these reviews as helpful; for reviews that gives 1-star, other users tend to vote these reviews as debatable.

After reviewing the structure and content of our data, we trimmed and sampled the original database:

Id	ProductId	UserId	Score	Summary	Text	SummartAndText	HelpfulPortion
321334	B008RWUHA6	A30JPZ9TZ7I61U	5	Tasty, healthy, and perfect for vegans ~	So despite this saying low-glycemic, it actual...	Tasty, healthy, and perfect for vegans ~So des...	1.0
543019	B00897UVVE	A3HEGKD16C4WZS	5	Authentic	This is the real deal, not some cheap coffee p...	AuthenticThis is the real deal, not some cheap...	1.0
299018	B004AUKLCI	A2QZUDPHCF6YRR	5	sweet nut and fruit gift	The item arrived as scheduled in excellent con...	sweet nut and fruit giftThe item arrived as sc...	1.0
357399	B001CJE4IC	A23TISZZ4Y937H	5	Nice large box	This was for my mother who will only chew Chic...	Nice large boxThis was for my mother who will ...	1.0
436554	B000CQ01JW	A3CC24EQUYSGGX	4	Grandson loves it...	I got an order of this for myself, since it ha...	Grandson loves it...I got an order of this for...	1.0

Figure 4. Dataset Samples.

We kept scores, corresponding review content, and helpfulness portion as training inputs and outputs in the models we were implementing next, while keeping the unique review id, product id, and user id for possible future data analysis works.

3. Model Implementation

3.1. Naïve word embedding + LSTM

Word embedding is a Natural Language Processing (NLP) technique that is used to represent words as vectors of numerical values. By setting a vector length that is related to the number of features we want to represent, we can map words to vectors in a high-dimensional space, and update vector values using a neural network structure. By using word embedding, we can capture the semantic relations between words. For example, 'cat' and 'dog' might be closer to each other than 'cat' and 'computer' in vector space.

Long Short-Term Memory (LSTM) is a type of recurrent neural network that is particularly useful in NLP tasks. LSTM is designed to process sequences of data, such as sequences of words in a sentence. It can remember information over a long period of time and selectively forget information that is no longer relevant, which makes it suitable for language modeling tasks such as semantic analysis of user reviews that may consist of hundreds of words.

Thus, the basic model we developed was to assemble an LSTM layer after word embeddings, expecting it to help improve the performance of NLP machine learning models.

3.2. Bert & RoBERTa

Bidirectional Transformers (Bert) [1] is a pretrained technique using the encoder section of the Transformer [2]. Shown in Figure.4, Bert first embeds the input tokens by pretrained embedding dictionary. Then the Transformer encoder section encodes word embeddings into latent state. According to the tasks, the final layer would map the hidden state to corresponding form of output. In our sentimental analysis tasks, which is essentially the 5-class text classification, the output layer is the fully connected layer with softmax (Fig.4) that can provide a 5-dimensional vector for training using the one-hot label.

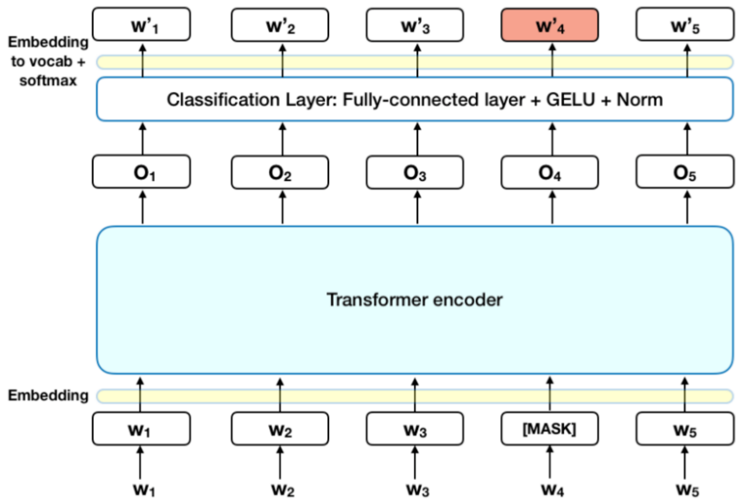

Figure 4. Illustration of Bert

Except for Bert, we also deployed RoBERTa as another baseline for large pretrained model[3]. It worth noting that the essential motivation of RoBERTa is like Bert in the domain of pretrained large language models. For better optimization results, RoBERTa replaced static masking with dynamic masking in the training phase. And altered character-level encoding to byte-level. Optimizations facilitate a more robust pretrained model and can handle more unseen words in upcoming tasks.

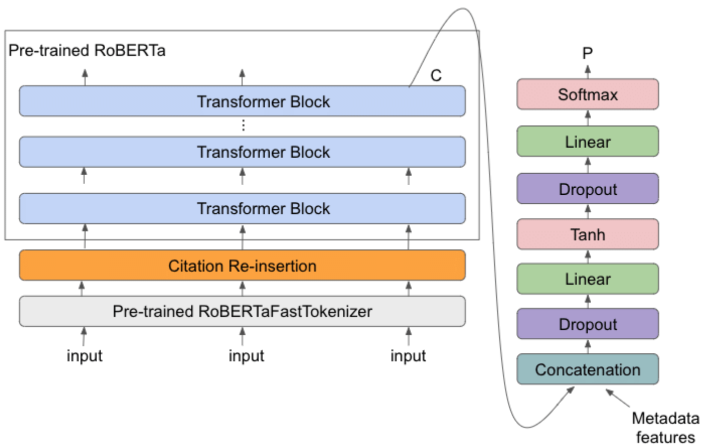

Figure 5. Illustration of Roberta

3.3. GPT

In the domain of pretrained large language models, Generative Pre-trained Transformer [4] also showed spectacular performance in various down streaming tasks, especially the text generation. However, different from bi-directional training strategy using masking in Bert and RoBERTa, GPT is an auto-regressive model, which is trained using the scheme to predict every next word based on previous words in unidirectional approach. It implies that GPT would be more capable of text generation tasks than other tasks. However, it can still be applied to our baseline experiments.

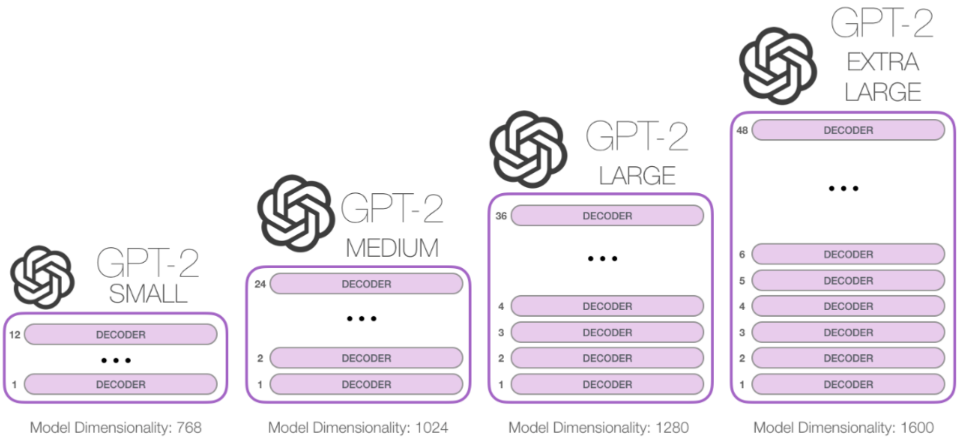

Figure 6. Illustration of GPT family.

3.4. Helpfulness Focal Loss

In this project, except for adopting cutting-edge models, we attempted to deploy some data-driven techniques for a better understanding of customers' reviews. Specifically, we considered the helpfulness index, which is illustrated in section 2, to leverage the dataset in our sentimental analysis. It is worth noting that helpfulness indicates the review of the review, higher helpfulness implies more valuable latent information hidden in the comments. Therefore, we want the model to emphasize more on these pieces of information during the training phase. We proposed a **helpfulness focal loss**, which is the weighted cross-entropy in accordance with every training batch (**Eq**.1). It will provide larger gradient and thus a better optimization result for valuable batches during the training phase.

$$\mathcal{L}_{cls} = \left(1 + \frac{1}{N}\sum_{i=0}^{N} h_i\right) * CE(y, \hat{y}) \tag{1}$$

3.5. Bert + GCN

Large-scale pretrained models such as BERT are highly effective in learning implicit but rich text semantics in language at scale. This is achieved through the use of deep neural networks and extensive training on massive amounts of data, allowing the model to gain a comprehensive understanding of the nuances of language. On the other hand, models like GCN utilize graph convolutional networks to consider the influence of neighboring instances in the decision-making process. This makes the model more robust against outliers and improves its ability to generalize new and unseen data. By incorporating both BERT and GCN, models can achieve a more comprehensive understanding of the data and make more accurate predictions. In short, these techniques are highly effective in enabling machines to process and comprehend large volumes of text data, which is critical for a wide range of applications such as natural language processing, sentiment analysis, and machine translation.

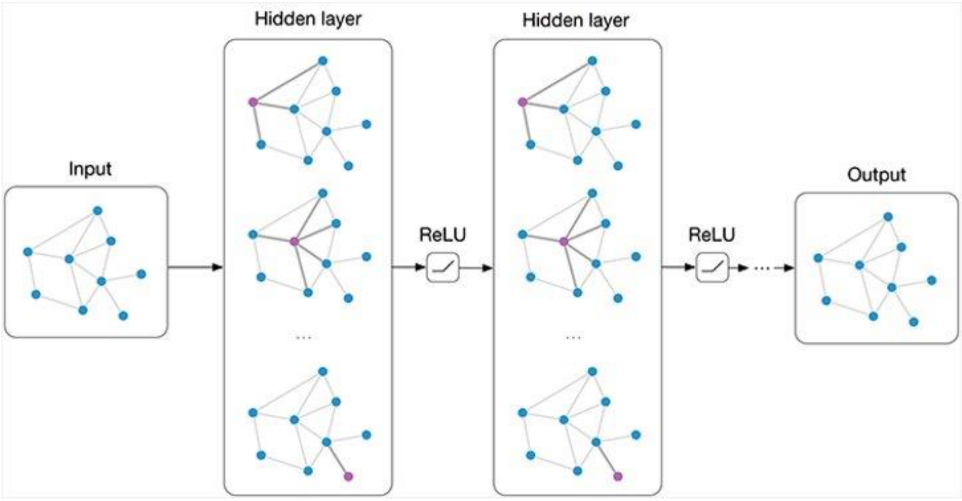

Figure 7. Illustration of GCN.

4. Evaluation Results

After training and tuning, we have collected the following results:

Method	Accuracy	Number of Parameters
LSTM	0.654	870,000
Bert	0.704	110,000,000
Bert + GCN	0.750	116,000,000
RoBERTa	0.759	123,000,000
RoBERTa + helpfulness	0.765	123,000,000
GPT2	0.719	1,500,000,000

Our naïve LSTM model achieved a 0.654 accuracy, which is the baseline model of our project. Bert and RoBERTa have improved the accuracy of prediction, with the tradeoff of training more parameters. Bert + GCN and RoBERTa + helpfulness has both achieved the best performance without feeding too many parameters to their corresponding basic models. As for GPT2, the explosion of parameter count did not result in a satisfying accuracy in exchange.

5. Conclusions

To explore and leverage information from the given dataset, we employed Natural Language Processing (NLP) techniques to perform a 5-class sentimental analysis. The traditional word embedding approach was compared with the more modern LLM approach, and it was found that the latter outperformed the former in terms of accuracy and efficiency. Our approach was designed based on this finding and further experiments were conducted to verify its effectiveness. The results showed that our approach achieved higher accuracy and lower error rates than the traditional word embedding approach. These findings suggest that the LLM approach is a promising method for sentiment analysis and can be used to extract meaningful insights from text data in a more efficient and accurate manner.

Github Repository: Our code is available at **https://github.com/Caffretro/Comp7103_Assignment1_data_mining_and_NLP.**

Reference

[1] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding." arXiv, May 24, 2019. doi: 10.48550/arXiv.1810.04805.

[2] A. Vaswani *et al.*, "Attention is All you Need," in *Advances in Neural Information Processing Systems*, Curran Associates, Inc., 2017. Accessed: Apr. 20, 2023. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html

[3] Y. Liu *et al.*, "RoBERTa: A Robustly Optimized BERT Pretraining Approach." arXiv, Jul. 26, 2019. doi: 10.48550/arXiv.1907.11692.

[4] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, "Language Models are Unsupervised Multitask Learners".