**Predicting Yelp Rating Polarity:**

**Leveraging NLP for Sentiment Analysis of Local Business Reviews**

Team 9 – Indranil Pal, Yasser Parambathkandy

**George Mason University**

**AIT526 Natural Language Processing**

**Dr. Heidari**

**06/28/2023**

**Abstract**

**The star-based ranking system used on platforms like Yelp.com has significant limitations in capturing the true essence of reviewers' sentiments and experiences. This approach fails to provide a comprehensive understanding of the nuanced aspects of reviews and relies on a single numerical value, lacking expressiveness and subjectivity. The subjective interpretation of star ratings creates ambiguity, as the same star rating may hold different meanings for different individuals. Sentiment analysis presents an invaluable opportunity to leverage the abundance of text-based data available on platforms like Yelp.com. While humans possess the innate ability to understand textual content, programs struggle with the complexity of interpreting large volumes of text swiftly. However, programs excel at processing vast amounts of data efficiently, making them ideal for analyzing and comprehending text-based reviews effectively. In response to these limitations, our project addresses the problem by leveraging sentiment analysis of review texts as a more effective and informative approach. This project recognizes the need to explore the richness and detailed information present in text-based reviews to provide a more holistic perspective. The main objectives of our project include employing NLP models focused on sentiment analysis to estimate the polarity of Yelp ratings based on the corresponding textual content. The aim is to conduct an in-depth exploration and evaluation of NLP algorithms, comparing their performance against pre-trained models. Additionally, a model with an ensemble approach that combines the strengths of DistilBERT and ELECTRA models, further enhancing the accuracy of sentiment analysis, has been developed and studied. The developed models provide a deeper understanding of reviewers' sentiments and enhance the evaluation process for businesses on platforms like Yelp.com.**

***Keywords — star-based ranking system, reviewers' sentiments, review text, subjectivity, sentiment analysis, Natural Language Processing (NLP), Recurrent Neural Networks (RNNs), LSTM, DistilBERT, ELECTRA, polarity estimation, ensemble approach***

**Introduction**

**One of the persisting challenges in modern applications revolves around the quantification of thoughts and opinions. Despite the widespread adoption of star-based ranking systems, they often fail to capture the true essence of a reviewer's sentiment. The subjective interpretation of star ratings creates ambiguity, where a five-star rating may hold vastly different meanings for different individuals. Consequently, it becomes crucial to explore the full-text reviews to obtain a more nuanced understanding. Our project aims to bridge this gap by enabling programs to analyze and comprehend the complete review text effectively.**

**Sentiment analysis presents an invaluable opportunity to leverage the abundance of text-based data available on the internet, which comprises a significant portion of unstructured information. While humans possess the innate ability to understand textual content, programs struggle with the complexity of interpreting large volumes of text swiftly. Conversely, programs excel at processing vast amounts of data efficiently, a task beyond human capabilities. Despite significant advancements, achieving absolute accuracy in sentiment analysis remains an ongoing challenge that fuels our project.**

**This project focuses exclusively on text-based reviews extracted from Yelp.com, a prominent platform that allows users to provide feedback on businesses using a five-star rating system alongside open-ended text reviews. By scrutinizing the textual content, the limitations of simplistic star ratings can be transcended, and a deeper understanding of reviewers' sentiments can be unraveled. This comprehensive analysis will enhance the evaluation process and provide a more holistic perspective on the quality of businesses.**

**Related Work**

**The various approaches and techniques in sentiment analysis, aiming to improve the accuracy and effectiveness of correlating user reviews with sentiment or ratings are as follows:**

**In their study, Yates et al. (2013) acknowledged the inherent noisiness of user reviews that are labeled with discrete numeric values. To mitigate this noise, they experimented with a different approach by categorizing the reviews simply as positive or negative, instead of assigning specific numeric values. Remarkably, this alternative labeling method achieved higher accuracy compared to traditional supervised learning algorithms that relied on discrete numeric values. Andrew L. Maas et al. (2011) employed sophisticated classification techniques, including both unsupervised and supervised methods, to learn word vectors that captured semantic term-document information and rich sentiment content. This approach enabled the development of a classifier that could leverage continuous and multidimensional sentiment information, leading to more comprehensive sentiment analysis. K. Yessenov et al. (2009) demonstrated that even simpler techniques, such as the bag-of-words model refined with carefully selected features based on the semantics and syntactic information from the text, can yield effective results. By applying these techniques, they developed a straightforward classifier with a high success rate. In the study conducted by F. Peleja et al. (2013), it was found that Support Vector Machines (SVMs) outperformed rule-based classifiers in accurately correlating user reviews with ratings. This highlights the effectiveness of SVMs as powerful tools for sentiment analysis. Additionally, the study emphasized the importance of exploring well-documented classifiers and assessing their efficacy in the specific context. V. Narayanan et al. (2013) conducted research demonstrating that well-tailored classifiers such as Naive Bayes hold significant potential when combined with specific techniques such as effective negation handling, word n-grams, and feature selection using mutual information. By optimizing these approaches, the results achieved with Naive Bayes alone reached approximately eighty-eight percent accuracy. It is worth noting, as highlighted by Leung et al. (2008), that sentiment analysis techniques are less effective without domain-specific knowledge. The effectiveness of analyzing sentiment varies depending on the specific field or domain, such as computer product reviews versus restaurant reviews.**

**This project extends related work: This project builds upon existing research by exploring Natural Language Processing (NLP) algorithms, specifically focusing on Recurrent Neural Networks (RNNs) such as LSTM. The objective is to compare the performance of these algorithms with pre-trained models like DistilBERT and ELECTRA (Efficiently Learning an Encoder that Classifies Token Replacements Accurately). ELECTRA aims to train text encoders as discriminators rather than generators, while DistilBERT is a method for pre-training a smaller general-purpose language representation model that can be fine-tuned for various tasks, similar to larger models. The implementation of these models will be supported by PyTorch packages and other relevant tools and technologies. By utilizing these resources, the goal is to develop a robust and effective model for the project.**

**Objectives**

**The main objectives of this project encompass the following aspects:**

1. **Employing NLP models focused on sentiment analysis to estimate the polarity of Yelp ratings for local business reviews based on the corresponding textual content.**
2. **Conducting an in-depth exploration and evaluation of NLP algorithms, particularly Recurrent Neural Networks (LSTM), while comparing their performance against pre-trained models such as DistilBERT and ELECTRA.**
3. **Creating an ensemble approach that incorporates DistilBERT and ELECTRA models. This involves individually inputting the training dataset into each model and subsequently combining their outputs using a linear layer to serve as a classifier.**
4. **Evaluate and compare the performance of all four models.**

**The Dataset**

**The dataset has been sourced from Kaggle, the Yelp reviews polarity dataset. This dataset is derived from the original dataset provided by the Yelp Dataset Challenge in 2015. Ratings of 1 and 2 were categorized as negative, labeled as 0, while ratings of 3 and 4 were considered positive, labeled as 1. Consequently, the dataset is structured for binary classification. It comprises a substantial amount of data, with over 560,000 training sample points and 38,000 testing sample points. Both the training and testing sample points are evenly distributed across both labels, ensuring a balanced representation. The feature column of the dataset corresponds to the review text, while the target column denotes the associated label.**

**Architecture**

**Two distinct solutions have been implemented for text analysis: an LSTM-based solution and a Transformer-based solution.**

**LSTM(Long Short-Term Memory) is a variant of RNN(Recurrent Neural Network) that address the vanishing gradient problem in RNNs. They introduce a memory cell that can selectively remember or forget information over time. In the "vanishing gradient" problem, the gradient signal diminishes over time, making it difficult to capture long-term dependencies. Yelp review sentiment analysis involves classifying reviews into positive or negative sentiments based on their content. RNNs, particularly LSTM networks, are well-suited for this task due to their ability to capture contextual information and long-term dependencies in sequential data. In the context of Yelp reviews, the sentiment of a particular word or phrase can depend on the preceding words or phrases. For example, the phrase "not bad" has a positive sentiment, while "not good" has a negative sentiment. RNNs can capture these dependencies by maintaining a hidden state that carries information from previous words, enabling the model to understand the sentiment in context. LSTM networks, with their memory cell and gating mechanisms, can effectively model long-range dependencies in reviews. They can remember important contextual information from earlier parts of a review and use it to inform the sentiment classification of subsequent words or phrases. This capability makes LSTMs a suitable choice for Yelp review sentiment analysis, allowing the model to better understand the overall sentiment expressed in a review and make accurate predictions.**

**LSTM Model layers**

1. **Embedding Layer - This layer converts input text indices into GloVe(Global Vectors for Word Representation) embeddings. GloVe embeddings are a word representation technique that captures semantic relationships between words based on co-occurrence patterns. They provide numerical representations of words that enable machines to understand and process textual data more effectively.**
2. **LSTM Layer - This layer takes the embedded input and processes it through the LSTM. The input size parameter is set to embedding dimension size, the hidden size parameter is set to 16 which determines the number of hidden units in each LSTM layer, and there are three LSTM layers.**
3. **Linear Layer - This layer acts as a learnable weighted average over time of all the LSTM outputs along the input sequence. It reshapes the LSTM output and applies a linear transformation to reduce the dimensionality to hidden size.**
4. **Batch Normalization Layer - This layer performs batch normalization along the second dimension of the input tensor. It normalizes the input tensor to have zero mean and unit variance.**
5. **Linear Layer - This layer maps the averaged representation obtained from the previous layer to the output classes. It transforms the input of size 16 to the output size of 2 (Yelp polarity is 1 or 0).**
6. **Dropout Layer - This layer applies dropout regularization to the input tensor. It randomly zeros some elements of the input tensor with probability 0.5.**

**The LSTM Pytorch Module definition is provided below -**

**LSTMModule(**

**(embedding): Embedding(660404, 50, padding\_idx=0)**

**(lstm): LSTM(50, 16, num\_layers=3, dropout=0.5)**

**(mean): Linear(in\_features=11200, out\_features=16, bias=True)**

**(bn\_mean): BatchNorm1d(16, eps=1e-05, momentum=0.1, affine=True, track\_running\_stats=True)**

**(out): Linear(in\_features=16, out\_features=2, bias=True)**

**(drop): Dropout(p=0.5, inplace=False)**

**)**

**LSTM Model Development Approach**

**The implementation of the LSTM model involves several steps as described below. The detailed steps of the implementation is provided below:**

1. **Data Preparation: The data is first cleaned by removing stopwords, punctuation, and lemmatize the words. The zero-length strings (empty after cleaning) are removed. The data is then split into training and validation steps. The above is repeated for the test data and saved.**
2. **Load GloVe embeddings: The GloVe embeddings file is loaded and a dictionary is created based on words from the review text.**
3. **Setting up device and random seed: The code determines the device for computation (GPU or CPU) and sets a random seed for reproducibility.**
4. **Defining hyperparameters: The hyperparameters for the LSTM model are set, including sequence length, embedding dimension, number of epochs, learning rate, batch size, hidden size, number of LSTM layers, and dropout rates.**
5. **Data preparation: The code loads the cleaned training, validation, and test data from CSV files. It processes the raw text sentences and labels, converts them into sequences of token IDs, and prepares them as PyTorch tensors.**
6. **LSTM model: The LSTM model for sentiment analysis is defined as described in the architecture section.**
7. **Training loop: The model is trained for the specified number of epochs. In each epoch, the model parameters are updated based on the computed loss and gradients. Training loss and accuracy are calculated and displayed.**
8. **Evaluation on the test set: After training, the model parameters are loaded, and the model is evaluated on the test set. The accuracy on the test set and the confusion matrix are calculated and printed.**

**Transformer Models**

**DistilBERT and ELECTRA (Efficiently Learning an Encoder that Classifies Token Replacements Accurately) are two advanced transformer-based models that have shown excellent performance in various natural language processing tasks, including sentiment analysis.**

**DistilBERT is a smaller and more efficient version of the popular BERT (Bidirectional Encoder Representations from Transformers) model. BERT revolutionized the field of natural language processing by introducing a pre-training and fine-tuning approach. For Yelp review sentiment analysis, DistilBERT is a good choice due to its ability to capture fine-grained contextual information. It understands the meaning and sentiment of words in the context of the entire review. By leveraging the pre-training on a vast amount of text data, DistilBERT effectively learn representations that capture the sentiment nuances present in Yelp reviews. It provides accurate sentiment predictions and handles the complexities of language used in user-generated content.**

**ELECTRA (Efficiently Learning an Encoder that Classifies Token Replacements Accurately) is another transformer-based model that has gained attention for its efficiency and strong performance. ELECTRA employs a novel approach called "discriminator training." where a small portion of input tokens is randomly replaced, and the model is trained to predict whether each token was replaced or not. This method forces the model to learn more efficient and accurate representations. ELECTRA's ability to capture fine-grained token-level information and understand the context of the replaced tokens can help it accurately infer sentiment in Yelp reviews. It can effectively leverage its pre-trained representations to grasp the sentiment nuances and provide reliable sentiment analysis results.**

**Transformer Model Development Approach**

**For the Transformer-based models, specifically DistilBERT and ELECTRA, a similar code structure is followed. The first step involves preprocessing the training data, divided into separate training and validation sets. Depending on the chosen model, either DistilBertTokenizerFast or ElectraTokenizerFast is used to process the data. Data loaders are created to handle the training and validation sets efficiently. Next, the selected Transformer model is initialized, ensuring proper configuration and setup. During training, the AdamW optimizer is employed along with a learning rate scheduler. The model is fed with data batches from the training loader, and the loss is calculated based on the model's outputs. The weights and biases of the model are updated accordingly. Finally, the performance of the model on the validation set is evaluated, and it is saved for future use.**

**To further enhance prediction accuracy, an ensemble approach is implemented by combining the strengths of both the DistilBERT and ELECTRA models. Individually, the training dataset is fed to each model, obtaining their respective outputs. These outputs are then passed through a linear layer acting as a classifier to merge the predictions. This ensemble technique leverages the unique perspectives and strengths of both models, resulting in improved prediction accuracy and overall robustness.**

**Proposed development platforms**

**The solution was developed in a Windows 11 WSL2 environment with specifications - AMD Ryzen 9 5950X 16-Core Processor 3.40 GHz, 64GB Memory, Nvidia GTX 3090 GPU, Cuda 12.1, cuDNN 12.1. The training was performed using GPUs. CPU based training was 30x slower. The program was developed in a conda environment utilizing software packages like nltk, pandas, pytorch, scikit-learn, transformers.**

**Results and Analysis**

**The performances results of the four models have been presented in Table 1.**

**Test Data Size: 37999**

**Train Data Size: 503973**

**Validation Data Size: 55998**

|  |  |  |  |
| --- | --- | --- | --- |
| **Model Name** | **Validation Accuracy** | **Test Data Accuracy** | **Time taken**  **(In minutes)** |
| **LSTM** | **93.35** | **93.48** | **20** |
| **DistilBert** | **92.47** | **93.01** | **116** |
| **ELECTRA** | **94.78** | **94.94** | **87** |
| **Ensemble** | **92.88** | **93.14** | **188** |

Table 1: Model Performance Comparison

**The LSTM model achieved a decent accuracy score on both the validation and test datasets. It is the fastest model, taking only 20 minutes for training. However, its accuracy is slightly lower compared to the other models. The DistilBERT model performed reasonably well, achieving a high accuracy score on the test dataset. However, its validation accuracy is lower compared to the other models. It took significantly longer to train compared to the LSTM model. The ELECTRA model demonstrated impressive performance with high accuracy scores on both the validation and test datasets. It outperforms the other models in terms of accuracy. Although it took longer to train compared to the LSTM model, it trained faster than the DistilBERT model. The ensemble model achieved competitive accuracy scores, similar to the DistilBERT and LSTM models. However, its validation accuracy is slightly higher. The ensemble model took the longest time to train, almost ten times longer than the LSTM model. Based on these findings, the ELECTRA model appears to be the most promising, considering its high accuracy and relatively fast training time compared to the ensemble model.**

**Conclusion**

**In conclusion, this project addressed the limitations of star-based ranking systems in capturing the true essence of reviewers' sentiments and experiences. By leveraging sentiment analysis of review texts, the project aimed to provide a more comprehensive understanding of the nuanced aspects of reviews on platforms like Yelp.com. The objectives of the project included employing NLP models focused on sentiment analysis, exploring the performance of Recurrent Neural Networks (LSTM) compared to pre-trained models like DistilBERT and ELECTRA, and developing an ensemble approach combining the strengths of both models. Through the implementation and evaluation of different models, including LSTM, DistilBERT, ELECTRA, and an ensemble model, the project demonstrated the effectiveness of leveraging NLP algorithms for sentiment analysis in the context of Yelp reviews. The findings highlight the potential of transformer-based models like DistilBERT and ELECTRA in sentiment analysis tasks. These models can effectively capture fine-grained contextual information and understand the complexities of user-generated content, leading to more accurate sentiment predictions. The developed models provide businesses and platforms like Yelp.com with a deeper understanding of reviewers' sentiments, enabling them to make more informed decisions and improve the evaluation process. By analyzing the text-based content of reviews, businesses can gain insights into the specific aspects that drive positive or negative sentiments, allowing them to address areas of improvement or reinforce their strengths.**

**Future Work**

**Future work can focus on expanding the evaluation to larger datasets and exploring other transformer-based models for sentiment analysis. Additionally, incorporating domain-specific knowledge and customizing the models for specific industries or niches can further improve their performance. Moreover, leveraging user demographics and preferences in conjunction with sentiment analysis could provide even more personalized and insightful evaluations for businesses.**

**References**

**Yelp, Inc. (2022). Retrieved from** <https://www.kaggle.com/datasets/ilhamfp31/yelp-review-dataset>

**Maas, Andrew & Daly, Raymond & Pham, Peter & Huang, Dan & Ng, Andrew & Potts, Christopher. (2011). Learning Word Vectors for Sentiment Analysis. 142-150.**

**Peleja, Filipa & Magalhães, João. (2013). Opinions in User Reviews: An Evaluation of Sentiment Analysis Techniques. 10.13140/2.1.3177.1206.**

**Leung, Cane & Chan, Stephen. (2008). Sentiment Analysis of Product Reviews.**

**Narayanan, V., Arora, I., &amp; Bhatia, A. (2013). Fast and accurate sentiment classification using an enhanced naive Bayes model. Intelligent Data Engineering and Automated Learning – IDEAL 2013, 194–201. doi:10.1007/978-3-642-41278-3\_24**

**Yang, Z. (2020). Sentiment analysis of movie reviews based on machine learning. 2020 2nd International Workshop on Artificial Intelligence and Education. doi:10.1145/3447490.3447491**

**Yates, A., Goharian, N., &amp; Yee, W. G. (2013). Semi-supervised probabilistic sentiment analysis: Merging labeled sentences with unlabeled reviews to identify sentiment. Proceedings of the American Society for Information Science and Technology, 50(1), 1–10. doi:10.1002/meet.14505001031**

**Appendix**

**A sample of the dataset from** <https://www.kaggle.com/datasets/ilhamfp31/yelp-review-dataset>

|  |  |
| --- | --- |
| Text | Label |
| Phone calls always go to voicemail and messages are not returned. Stupid way to do business. | 1 |
| Ryan Rocks! I called him this morning for some sprinkler help and some potential landscaping for my backyard. He showed up to my house within an hour and answered all of my questions! When I showed him how my irrigation box was leaking, he knew right away what to do - he went a grabbed a new part out of his truck, put it on, and it was fixed! He then set up my watering timer and we talked about some landscaping and pricing. His prices seemed great and I will definitely use him in the future! To top it all off, when I asked him what I owed him for fixing my irrigation leak and his time, he said don't worry about it! I was shocked and couldn't believe his service! Thank you so much Ryan! | 2 |
| We tried the Cheese Danish w lemon. Creamy, flaky and not overly sweet. It was fantastic. Chocolate croissant was yummy too! | 2 |
| Long line, inefficient staff. Maybe my expectations were too high but it just wasn't as good as I was hoping for the calories. We had the bread pudding and carrot cake cookie. Eh. | 1 |